From:

Snapshot-Content-Location: http://localhost:7860/?lang=bash&view=api Subject: Gradio Date: Tue, 9 Dec 2025

22:18:10 +0800 MIME-Version: 1.0 Content-Type: multipart/related; type="text/html"; boundary="----

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI----" ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/html Content-ID:

Content-Transfer-

Encoding: quoted-printable Content-Location: http://localhost:7860/?lang=bash&view=api

=09

=09

=09 =09 =09 =09 =

Gradio= <= link rel=3D"stylesheet" crossorigin=3D"" href=3D"http://localhost:7860/asse= ts/Index-BdvHG3t0.css">

<= /div>

**F5-TTS Demo Space**

This is a local web UI for

F5-TTS

with advanced batc= h processing support. This app supports the following TTS

models:

•

F5-TTS

(A Fairytaler that Fakes Fluent and Faithful= Speech with Flow Matching)

•

E2 TTS

(Embarrassingly Easy Fully Non-Autoregressiv= e Zero-Shot TTS)

The checkpoints currently support English and Chinese.

If you're having issues, try converting your reference audio to WAV or M= P3, clipping it to 12s with =E2=9C=82 in the

bottom right corner (otherwi= se might have non-optimal auto-trimmed result).

**NOTE: Reference text will be automatically transcribed with Whis= per if not provided. For best results, keep**

**your reference clips short (<= ;12s). Ensure the audio is fully uploaded before generating.**

Choose TTS Model

=

F5-TTS\_v1

=

E2-TTS

<= span

class=3D"svelte-1bx8sav">Custom

Basic-T= TS

Multi-Speech

Voice-Chat

= Credits

Basic-TTS

Multi-Speech

Voice-Chat

Credits

=

<= svg xmlns=3D"http://www.w3.org/2000/svg" width=3D"100%" height=3D"100%" vie= wBox=3D"0 0 24 24"

fill=3D"none" stroke=3D"currentColor" stroke-width=3D"1.= 5" stroke-linecap=3D"round" stroke-linejoin=3D"round"

class=3D"feather feat= her-music">

= Reference Audio

=E5=B0=86=E9=

=9F=B3=E9=A2=91=E6=8B=96=E6=94=BE=E5=88=B0=E6=AD=A4=E5=A4=84

- =E6=88=96 -

=E7=82=B9=E5=87=BB=E4=B8=8A=E4=BC= =A0

3D"F5-TTS\_v1"

3D"E2-TTS"

3D"Custom"

<= button class=3D"icon svelte-snayfm selected" aria-label=3D"Upload file">

Text to Generate

<= div id=3D"component-5" class=3D"block svelte-1svsvh2" dir=3D"ltr" style=3D= "--start-top: 0px; --start-left:

0px; --start-width: 0px; --start-height: 0= px; border-style: dashed; overflow: hidden; flex-grow: 1; min-width: min(16=

0px, 100%); border-width: var(--block-border-width);">

=

Load Text to Generate fr= om File (.txt)

=

=

=E5=B0=86=E6=96=87=E4=BB=B6=E6=8B=96=E6=94=BE=E5=88=B0=E6=AD=A4=E5=A4=84

- =E6=88=96 -

=E7=82=B9=E5=87=BB=E4= =B8=8A=E4=BC=A0

= Synthesize

Advanced Settings

=E2=96=BC

=

Reference Text

=

Leave blank to automatica= lly transcribe the reference audio. If you enter text or upload a file, it = will override

automatic transcription.

<= /label>

Loa= d Reference Text from File (.txt)

=E5=B0=86=E6=96=87=E4=BB=B6=E6=8B=96=E6=94=BE=E5=88=B0=E6=

=AD=A4=E5=A4=84

- =E6=88=96 -

=E7= =82=B9=E5=87=BB=E4=B8=8A=E4=BC=A0

Randomize Seed

<= /label>

Check to use a random seed for each generation. Uncheck to use = the seed specified.

=E6=95=B0=E5=AD=97

Remove Silences

If undesired long silen= ce(s) produced, turn on to automatically detect and crop.

Speed

Adjust the speed of the= audio.

=E2=86=BA

0.3

=

2

=

NFE Steps

<= div class=3D"svelte-j9uq24">

=

Set the number of denoising steps.

=E2=86=BA

4

=

64

=

Cross-Fade Duratio= n (s)

Set the duration of the cross-fade between audio clips.

=E2=86=BA

0

1

Synthesized Audio

Spectrogram

**Multipl= e Speech-Type Generation**

This section allows you to generate multiple speech types or multiple pe= ople's voices. Enter your text in the format

shown below, or upload a .txt = file with the same format. The system will generate speech using the approp= riate type.

If unspecified, the model will use the regular speech type. The= current speech type will be used until the next speech

type is specified.<= /p>

**Example Input:=**

{Regular} Hello, I'd like to order a sandwich please.

{Surprised} What do you mean you're out of bread?

{Sad} I really wanted a sandwich though...

{Angry} You know what, darn you and your little shop!

{Whisper} I'll just go back home and cry now.

{Shouting} Why me?!

**Example Inp= ut 2:**

{"name": "Speaker1\_Happy", "seed": -1, "speed": 1} Hello, I'd like to order= a sandwich please.

{"name": "Speaker2\_Regular", "seed": -1, "speed": 1} Sorry, we're out of br= ead.

{"name": "Speaker1\_Sad", "seed": -1, "speed": 1} I really wanted a sandwich= though...

{"name": "Speaker2\_Whisper", "seed": -1, "speed": 1} I'll give you the last= one I was hiding.

Upload diffe= rent audio clips for each speech type. The first speech type is mandatory. = You can add additional speech

types by clicking the "Add Speech Type" butto= n.

Speech Type Name

Inser= t Label

Regular Reference Audio

=E5=B0=86=E9=9F=B3=E9=A2=91=E6=8B=96=E6=94=BE=

=E5=88=B0=E6=AD=A4=E5=A4=84

- =E6=88=96 -=

=E7=82=B9=E5=87=BB=E4=B8=8A=E4=BC=A0

=

<= div class=3D"form svelte-1vd8eap" style=3D"flex-grow: 0; min-width: min(0px= , 100%);">

Reference Text (Regular= )

=E6=BB=91=E5=9D=97

Seed, -1 for random

=E2=86=BA

-1

999

=E6=BB=91=E5=9D=97

Adjust the speed

=E2=86=BA

0.3

2

=

<= div id=3D"component-47" class=3D"block svelte-1svsvh2 auto-margin" dir=3D"l= tr" style=3D"--start-top: 0px; --

start-left: 0px; --start-width: 0px; --sta= rt-height: 0px; border-style: dashed; overflow: hidden; min-width: min(160p= x,

100%); border-width: var(--block-border-width);">

<= /div>

Load Reference Text from Fil= e (.txt)

=E5=B0=

=86=E6=96=87=E4=BB=B6=E6=8B=96=E6=94=BE=E5=88=B0=E6=AD=A4=E5=A4=84

- =E6=88=96 -

=E7=82=B9=E5=87=BB=E4=B8=8A= =E4=BC=A0

=

Add Speech Type

Text to Generate

<= /div>

Load Text to Generate from F= ile (.txt)

=E5=B0=

=86=E6=96=87=E4=BB=B6=E6=8B=96=E6=94=BE=E5=88=B0=E6=AD=A4=E5=A4=84

- =E6=88=96 -

=E7=82=B9=E5=87=BB=E4=B8=8A= =E4=BC=A0

=

Advanced Settings

=E2=96=BC

Show Cherry-pick Interface

Turn o= n to show interface, picking seeds from previous generations.

Remove Silences

Turn on to automa= tically detect and crop long silences.

Generate Multi-Style Spe= ech

Synthesized Audio<= /label>

**Voice Chat**

Have a conversation with an AI using your reference voice!

1.

Upload a reference audio clip and optionally its transcript (via text o= r .txt file).

2.

Load the chat model.

3.

Record your message through your microphone or type it.

4.

The AI will respond using the reference voice.



Load Chat Model

**Credits**

**•**

**mrfakename**

**for the original**

**online demo**

**•**

**RootingInLoad**

**for initial chunk generation and**

**podc= ast app exploration**

**•**

**jpgallegoar**

**for multiple speech-type generation &=**

**voice chat**

=E9=80=9A=E8=BF=87 API =E4=BD=BF=E7=94=A8

3D"=E6=A0=87=E5=BF=97"

=C2=B7

=E4=BD=BF=E7=94=A8

Gradio =E6=9E=84=E5=BB=BA

3D"=E6=A0=87=

=C2=B7

Stop Re= cording

3D"Stop

=C2=B7

=E8=AE=BE=E7=BD=AE

3D"=E8=AE=BE=E7=BD=AE"

**3D""**

**AP= I documentation**

**http://localhost:7860/**

API = Recorder

**430**

**API = endpoints**

Choose one of th= e following ways to interact with the API.

•

3D""

Python

•

3D""

JavaScript

•

=3D""

cURL

•

3D""

MCP

1. Confirm that you have cURL insta= lled on your system.

copy

$ curl --version

2. Find the API endpoint below corresponding to your de= sired function in the app. Copy the code snippet, replacing the

placeholder values with your own input data. =09 Or use the

API Recorder

to automatically generate your API requests.

Making a prediction and getting a result requires

**2 requests**

: a

POST

and a

GET

request. The

POST

request returns an

EVENT\_ID

, which is= used in the second

GET

request to fetch the resu= lts. In these snippets, we've used

awk

and =

read

to parse the results, combining these two requests into one command for ease of use. See

curl docs

.

**API name:**

**/switch\_tts\_model**

copy

curl -X POST http://localhost:7860/gradio\_= api/call/switch\_tts\_model -s -H "Content-Type:

application/json" -d '{ "data": [ "F5-TTS\_v1" ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/switch\_tts\_= model/$EVENT\_ID

**Accepts 1 parameter:**

[0]

s= tring

Required

The input value that is provided in the "Choose T= TS Model" Radio component.

**Returns list of 3 elements**

[0]

string

The output value that appears in the "Model: local\_path | hf://user\_id= /repo\_id/model\_ckpt" Dropdown component.

[1]

string

The output value that appears in the "Vocab: local\_path | hf://u= ser\_id/repo\_id/vocab\_file" Dropdown component.

[2]

string

The output value that appears in the "Config: in a dictionary fo= rm" Dropdown component.

**API name:**

**/set\_custom\_model**

copy

curl -X POST http://localhost:7860/gradio\_= api/call/set\_custom\_model -s -H "Content-Type:

application/json" -d '{ "data": [ "hf://SWivid/F5-TTS/F5TTS\_v1\_Base/model\_1250000.safetensors",

"hf://SWivid/F5-TTS/F5TTS\_v1\_Base/vocab.txt", "{"dim": 1024, "depth": 22, "heads": 16, "ff\_mult": 2,

"text\_dim": 5= 12, "conv\_layers": 4}" ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/set\_custom\_= model/$EVENT\_ID

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Model: = local\_path | hf://user\_id/repo\_id/model\_ckpt" Dropdown component.

[1]

string<= /span>

Required

The input value that is provided in the "Vocab: local\_p= ath | hf://user\_id/repo\_id/vocab\_file" Dropdown component.

[2]

string<= /span>

Required

The input value that is provided in the "Config: in a d= ictionary form" Dropdown component.

**Returns 1 element**

**API name:**

**/set\_custom\_model\_1**

copy

curl -X POST http://localhost:7860/gradio\_= api/call/set\_custom\_model\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ "hf://SWivid/F5-TTS/F5TTS\_v1\_Base/model\_1250000.safetensors",

"hf://SWivid/F5-TTS/F5TTS\_v1\_Base/vocab.txt", "{"dim": 1024, "depth": 22, "heads": 16, "ff\_mult": 2,

"text\_dim": 5= 12, "conv\_layers": 4}" ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/set\_custom\_= model\_1/$EVENT\_ID

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Model: = local\_path | hf://user\_id/repo\_id/model\_ckpt" Dropdown component.

[1]

string<= /span>

Required

The input value that is provided in the "Vocab: local\_p= ath | hf://user\_id/repo\_id/vocab\_file" Dropdown component.

[2]

string<= /span>

Required

The input value that is provided in the "Config: in a d= ictionary form" Dropdown component.

**Returns 1 element**

**API name:**

**/set\_custom\_model\_2**

copy

curl -X POST http://localhost:7860/gradio\_= api/call/set\_custom\_model\_2 -s -H "Content-Type:

application/json" -d '{ "data": [ "hf://SWivid/F5-TTS/F5TTS\_v1\_Base/model\_1250000.safetensors",

"hf://SWivid/F5-TTS/F5TTS\_v1\_Base/vocab.txt", "{"dim": 1024, "depth": 22, "heads": 16, "ff\_mult": 2,

"text\_dim": 5= 12, "conv\_layers": 4}" ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/set\_custom\_= model\_2/$EVENT\_ID

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Model: = local\_path | hf://user\_id/repo\_id/model\_ckpt" Dropdown component.

[1]

string<= /span>

Required

The input value that is provided in the "Vocab: local\_p= ath | hf://user\_id/repo\_id/vocab\_file" Dropdown component.

[2]

string<= /span>

Required

The input value that is provided in the "Config: in a d= ictionary form" Dropdown component.

**Returns 1 element**

**API name:**

**/load\_text\_from\_file**

<= button class=3D"sm secondary svelte-o34uqh" id=3D""> copy

<= div>

curl -X POST http://localhost:7860/gradio= \_api/call/load\_text\_from\_file -s -H "Content-Type:

application/json" -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Text= to Generate from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/load\_text\_from\_file\_1**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_1 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_1/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda**

copy

curl -X POST http://localhost:7860/gradio\_api/call/la= mbda -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda/$EVE= NT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/basic\_tts**

**Total requ= ests: 3 (**

**100%**

**successful)  |  p50/p90/p99: 15.98 s / 32.58 s /**

**36.31 s**

copy

curl -X POST http://localhost:7860/gradio\_api/call/basic\_t= ts -s -H "Content-Type: application/json" -d

'{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!", "Hello!!", true, true, 3, 0, 4,

0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/basic\_tts/$= EVENT\_ID

Accepts 9 parameters:

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Reference= Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[1]

string

Required

[2]

string<= /span>

Required

The input value that is provided in the "Text to Genera= te" Textbox component.

[3]

boolean=

Required

The input value that is provided in the "Remove Silenc= es" Checkbox component.

[4]

boolean=

Required

The input value that is provided in the "Randomize See= d" Checkbox component.

[5]

any

Required

The input value that is provided in the "parameter\_15" Num= ber component.

[6]

number<= /span>

Required

The input value that is provided in the "Cross-Fade Dur= ation (s)" Slider component.

[7]

number<= /span>

Required

The input value that is provided in the "NFE Steps" Sli= der component.

[8]

number<= /span>

Required

The input value that is provided in the "Speed" Slider component.

**Returns list of 4 elements**

[0]

T= he output value that appears in the "Synthesized Audio" Audio component.

[1]

string

The output value that appears in the "Spectrogram" Image component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "value\_15" Number component.

**API name:**

**/lambda\_1**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_1/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_2**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_2 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_2/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_1**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_1/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_1\_1**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/load\_text\_from\_file\_1\_1 -s -H "Content-Type:

application/json= " -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_1\_1/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_2**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_2 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_2/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_2**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_2 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_2/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_3**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_3 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_3/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_3**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_3 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_3/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_4**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_4 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_4/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_4**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_4 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_4/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_5**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_5 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_5/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_5**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_5 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_5/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_6**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_6 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_6/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_6**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_6 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_6/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_7**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_7 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_7/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_7**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_7 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_7/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_8**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_8 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_8/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_8**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_8 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_8/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_9**

copy

curl -X POST http://localhost:7860/gradio\_api/call/l= ambda\_9 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_9/$E= VENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_9**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/load\_text\_from\_file\_9 -s -H "Content-Type:

application/json" -d= '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_9/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_10**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_10 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_10/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_10**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_10 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_10/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_11**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_11 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_11/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_11**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_11 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_11/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_12**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_12 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_12/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_12**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_12 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_12/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_13**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_13 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_13/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_13**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_13 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_13/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_14**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_14 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_14/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_14**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_14 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_14/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_15**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_15 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_15/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_15**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_15 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_15/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_16**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_16 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_16/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_16**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_16 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_16/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_17**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_17 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_17/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_17**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_17 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_17/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_18**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_18 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_18/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_18**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_18 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_18/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_19**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_19 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_19/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_19**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_19 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_19/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_20**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_20 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_20/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_20**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_20 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_20/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_21**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_21 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_21/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_21**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_21 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_21/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_22**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_22 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_22/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_22**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_22 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_22/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_23**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_23 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_23/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_23**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_23 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_23/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_24**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_24 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_24/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_24**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_24 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_24/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_25**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_25 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_25/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_25**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_25 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_25/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_26**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_26 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_26/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_26**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_26 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_26/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_27**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_27 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_27/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_27**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_27 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_27/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_28**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_28 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_28/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_28**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_28 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_28/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_29**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_29 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_29/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_29**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_29 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_29/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_30**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_30 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_30/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_30**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_30 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_30/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_31**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_31 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_31/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_31**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_31 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_31/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_32**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_32 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_32/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_32**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_32 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_32/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_33**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_33 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_33/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_33**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_33 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_33/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_34**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_34 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_34/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_34**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_34 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_34/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_35**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_35 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_35/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_35**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_35 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_35/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_36**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_36 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_36/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_36**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_36 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_36/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_37**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_37 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_37/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_37**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_37 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_37/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_38**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_38 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_38/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_38**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_38 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_38/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_39**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_39 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_39/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_39**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_39 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_39/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_40**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_40 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_40/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_40**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_40 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_40/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_41**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_41 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_41/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_41**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_41 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_41/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_42**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_42 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_42/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_42**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_42 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_42/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_43**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_43 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_43/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_43**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_43 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_43/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_44**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_44 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_44/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_44**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_44 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_44/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_45**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_45 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_45/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_45**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_45 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_45/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_46**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_46 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_46/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_46**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_46 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_46/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_47**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_47 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_47/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_47**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_47 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_47/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_48**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_48 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_48/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_48**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_48 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_48/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_49**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_49 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_49/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_49**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_49 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_49/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_50**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_50 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_50/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_50**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_50 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_50/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_51**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_51 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_51/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_51**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_51 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_51/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_52**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_52 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_52/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_52**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_52 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_52/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_53**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_53 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_53/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_53**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_53 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_53/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_54**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_54 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_54/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_54**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_54 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_54/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_55**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_55 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_55/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_55**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_55 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_55/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_56**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_56 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_56/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_56**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_56 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_56/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_57**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_57 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_57/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_57**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_57 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_57/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_58**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_58 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_58/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_58**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_58 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_58/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_59**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_59 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_59/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_59**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_59 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_59/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_60**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_60 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_60/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_60**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_60 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_60/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_61**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_61 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_61/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_61**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_61 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_61/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_62**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_62 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_62/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_62**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_62 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_62/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_63**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_63 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_63/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_63**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_63 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_63/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_64**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_64 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_64/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_64**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_64 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_64/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_65**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_65 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_65/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_65**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_65 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_65/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_66**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_66 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_66/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_66**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_66 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_66/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_67**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_67 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_67/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_67**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_67 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_67/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_68**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_68 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_68/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_68**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_68 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_68/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_69**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_69 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_69/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_69**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_69 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_69/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_70**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_70 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_70/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_70**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_70 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_70/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_71**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_71 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_71/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_71**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_71 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_71/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_72**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_72 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_72/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_72**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_72 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_72/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_73**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_73 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_73/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_73**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_73 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_73/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_74**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_74 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_74/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_74**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_74 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_74/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_75**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_75 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_75/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_75**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_75 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_75/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_76**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_76 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_76/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_76**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_76 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_76/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_77**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_77 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_77/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_77**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_77 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_77/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_78**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_78 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_78/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_78**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_78 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_78/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_79**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_79 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_79/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_79**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_79 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_79/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_80**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_80 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_80/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_80**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_80 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_80/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_81**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_81 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_81/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_81**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_81 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_81/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_82**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_82 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_82/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_82**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_82 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_82/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_83**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_83 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_83/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_83**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_83 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_83/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_84**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_84 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_84/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_84**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_84 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_84/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_85**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_85 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_85/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_85**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_85 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_85/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_86**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_86 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_86/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_86**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_86 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_86/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_87**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_87 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_87/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_87**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_87 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_87/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_88**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_88 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_88/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_88**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_88 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_88/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_89**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_89 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_89/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_89**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_89 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_89/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_90**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_90 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_90/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_90**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_90 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_90/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_91**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_91 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_91/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_91**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_91 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_91/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_92**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_92 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_92/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_92**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_92 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_92/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_93**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_93 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_93/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_93**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_93 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_93/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_94**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_94 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_94/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_94**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_94 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_94/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_95**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_95 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_95/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_95**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_95 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_95/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_96**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_96 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_96/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_96**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_96 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_96/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_97**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_97 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_97/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_97**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_97 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_97/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_98**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_98 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_98/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_98**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_98 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_98/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/lambda\_99**

copy

curl -X POST http://localhost:7860/gradio\_api/call/= lambda\_99 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_99/$= EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

string

The output value that appears in the "Reference Text" Textbox component.

[1]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/load\_text\_from\_file\_99**

copy

curl -X POST http://localhost:7860/gra= dio\_api/call/load\_text\_from\_file\_99 -s -H "Content-Type:

application/json" = -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_99/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/add\_speech\_type\_fn**

copy

curl -X POST http://localhost:7860/gradio\_= api/call/add\_speech\_type\_fn -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/add\_speech\_= type\_fn/$EVENT\_ID

**Accepts 0 parameters:**

**Returns 1 element**

**API name:**

**/delete\_speech\_type\_fn**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/delete\_speech\_type\_fn -s -H "Content-Type:

application/json" -d= '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_1**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_1 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_1/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_2**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_2 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_2/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_3**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_3 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_3/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_4**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_4 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_4/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_5**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_5 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_5/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_6**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_6 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_6/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_7**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_7 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_7/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_8**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_8 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_8/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_9**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/delete\_speech\_type\_fn\_9 -s -H "Content-Type:

application/json= " -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_9/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_10**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_10 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_10/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_11**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_11 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_11/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_12**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_12 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_12/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_13**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_13 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_13/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_14**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_14 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_14/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_15**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_15 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_15/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_16**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_16 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_16/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_17**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_17 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_17/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_18**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_18 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_18/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_19**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_19 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_19/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_20**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_20 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_20/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_21**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_21 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_21/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_22**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_22 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_22/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_23**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_23 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_23/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_24**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_24 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_24/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_25**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_25 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_25/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_26**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_26 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_26/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_27**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_27 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_27/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_28**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_28 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_28/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_29**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_29 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_29/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_30**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_30 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_30/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_31**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_31 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_31/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_32**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_32 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_32/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_33**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_33 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_33/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_34**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_34 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_34/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_35**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_35 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_35/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_36**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_36 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_36/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_37**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_37 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_37/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_38**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_38 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_38/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_39**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_39 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_39/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_40**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_40 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_40/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_41**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_41 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_41/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_42**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_42 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_42/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_43**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_43 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_43/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_44**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_44 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_44/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_45**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_45 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_45/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_46**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_46 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_46/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_47**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_47 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_47/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_48**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_48 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_48/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_49**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_49 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_49/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_50**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_50 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_50/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_51**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_51 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_51/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_52**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_52 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_52/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_53**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_53 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_53/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_54**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_54 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_54/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_55**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_55 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_55/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_56**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_56 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_56/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_57**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_57 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_57/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_58**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_58 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_58/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_59**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_59 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_59/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_60**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_60 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_60/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_61**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_61 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_61/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_62**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_62 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_62/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_63**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_63 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_63/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_64**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_64 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_64/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_65**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_65 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_65/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_66**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_66 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_66/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_67**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_67 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_67/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_68**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_68 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_68/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_69**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_69 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_69/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_70**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_70 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_70/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_71**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_71 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_71/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_72**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_72 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_72/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_73**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_73 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_73/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_74**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_74 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_74/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_75**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_75 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_75/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_76**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_76 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_76/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_77**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_77 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_77/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_78**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_78 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_78/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_79**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_79 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_79/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_80**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_80 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_80/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_81**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_81 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_81/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_82**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_82 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_82/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_83**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_83 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_83/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_84**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_84 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_84/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_85**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_85 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_85/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_86**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_86 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_86/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_87**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_87 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_87/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_88**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_88 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_88/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_89**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_89 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_89/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_90**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_90 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_90/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_91**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_91 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_91/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_92**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_92 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_92/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_93**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_93 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_93/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_94**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_94 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_94/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_95**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_95 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_95/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_96**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_96 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_96/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_97**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_97 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_97/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/delete\_speech\_type\_fn\_98**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/delete\_speech\_type\_fn\_98 -s -H "Content-Type:

application/js= on" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/delete\_spee= ch\_type\_fn\_98/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 4 elements**

[0]

string

The output value that appears in the "Speech Type Name" Textbox component.

[1]

The output value that appears in the "Reference Audio" Audio component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

The output value that appears in the "Load Reference Text from File (.= txt)" File component.

**API name:**

**/insert\_speech\_type\_fn**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/insert\_speech\_type\_fn -s -H "Content-Type:

application/json" -d= '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_42" = Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_43" = Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_1**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_1 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_1/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_59" = Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_60" = Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_2**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_2 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_2/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_76" = Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_77" = Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_3**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_3 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_3/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_93" = Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_94" = Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_4**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_4 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_4/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_110"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_111"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_5**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_5 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_5/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_127"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_128"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_6**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_6 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_6/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_144"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_145"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_7**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_7 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_7/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_161"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_162"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_8**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_8 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_8/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_178"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_179"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_9**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/insert\_speech\_type\_fn\_9 -s -H "Content-Type:

application/json= " -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_9/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_195"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_196"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_10**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_10 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_10/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_212"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_213"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_11**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_11 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_11/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_229"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_230"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_12**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_12 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_12/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_246"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_247"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_13**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_13 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_13/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_263"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_264"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_14**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_14 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_14/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_280"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_281"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_15**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_15 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_15/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_297"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_298"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_16**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_16 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_16/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_314"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_315"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_17**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_17 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_17/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_331"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_332"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_18**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_18 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_18/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_348"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_349"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_19**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_19 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_19/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_365"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_366"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_20**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_20 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_20/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_382"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_383"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_21**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_21 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_21/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_399"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_400"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_22**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_22 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_22/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_416"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_417"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_23**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_23 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_23/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_433"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_434"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_24**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_24 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_24/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_450"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_451"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_25**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_25 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_25/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_467"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_468"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_26**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_26 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_26/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_484"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_485"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_27**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_27 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_27/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_501"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_502"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_28**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_28 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_28/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_518"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_519"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_29**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_29 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_29/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_535"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_536"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_30**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_30 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_30/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_552"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_553"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_31**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_31 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_31/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_569"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_570"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_32**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_32 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_32/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_586"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_587"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_33**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_33 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_33/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_603"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_604"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_34**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_34 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_34/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_620"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_621"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_35**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_35 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_35/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_637"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_638"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_36**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_36 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_36/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_654"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_655"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_37**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_37 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_37/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_671"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_672"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_38**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_38 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_38/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_688"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_689"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_39**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_39 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_39/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_705"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_706"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_40**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_40 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_40/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_722"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_723"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_41**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_41 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_41/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_739"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_740"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_42**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_42 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_42/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_756"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_757"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_43**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_43 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_43/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_773"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_774"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_44**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_44 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_44/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_790"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_791"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_45**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_45 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_45/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_807"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_808"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_46**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_46 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_46/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_824"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_825"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_47**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_47 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_47/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_841"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_842"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_48**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_48 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_48/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_858"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_859"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_49**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_49 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_49/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_875"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_876"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_50**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_50 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_50/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_892"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_893"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_51**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_51 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_51/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_909"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_910"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_52**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_52 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_52/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_926"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_927"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_53**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_53 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_53/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_943"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_944"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_54**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_54 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_54/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_960"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_961"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_55**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_55 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_55/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_977"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_978"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_56**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_56 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_56/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_994"= Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_995"= Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_57**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_57 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_57/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1011= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1012= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_58**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_58 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_58/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1028= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1029= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_59**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_59 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_59/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1045= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1046= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_60**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_60 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_60/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1062= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1063= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_61**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_61 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_61/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1079= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1080= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_62**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_62 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_62/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1096= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1097= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_63**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_63 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_63/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1113= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1114= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_64**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_64 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_64/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1130= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1131= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_65**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_65 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_65/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1147= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1148= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_66**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_66 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_66/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1164= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1165= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_67**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_67 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_67/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1181= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1182= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_68**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_68 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_68/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1198= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1199= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_69**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_69 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_69/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1215= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1216= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_70**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_70 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_70/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1232= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1233= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_71**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_71 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_71/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1249= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1250= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_72**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_72 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_72/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1266= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1267= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_73**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_73 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_73/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1283= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1284= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_74**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_74 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_74/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1300= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1301= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_75**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_75 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_75/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1317= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1318= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_76**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_76 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_76/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1334= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1335= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_77**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_77 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_77/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1351= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1352= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_78**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_78 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_78/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1368= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1369= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_79**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_79 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_79/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1385= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1386= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_80**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_80 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_80/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1402= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1403= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_81**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_81 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_81/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1419= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1420= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_82**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_82 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_82/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1436= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1437= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_83**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_83 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_83/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1453= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1454= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_84**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_84 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_84/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1470= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1471= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_85**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_85 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_85/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1487= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1488= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_86**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_86 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_86/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1504= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1505= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_87**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_87 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_87/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1521= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1522= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_88**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_88 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_88/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1538= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1539= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_89**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_89 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_89/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1555= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1556= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_90**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_90 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_90/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1572= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1573= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_91**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_91 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_91/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1589= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1590= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_92**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_92 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_92/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1606= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1607= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_93**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_93 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_93/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1623= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1624= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_94**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_94 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_94/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1640= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1641= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_95**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_95 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_95/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1657= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1658= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_96**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_96 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_96/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1674= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1675= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_97**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_97 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_97/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1691= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1692= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_98**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_98 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_98/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1708= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1709= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/insert\_speech\_type\_fn\_99**

copy

curl -X POST http://localhost:7860/g= radio\_api/call/insert\_speech\_type\_fn\_99 -s -H "Content-Type:

application/js= on" -d '{ "data": [ "Hello!!", "Hello!!", -1, 0.3 ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/insert\_spee= ch\_type\_fn\_99/$EVENT\_ID

**Accepts 4 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= string

Required

The input value that is provided in the "Text to= Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

number<= /span>

Required

The input value that is provided in the "parameter\_1725= " Slider component.

[3]

number<= /span>

Required

The input value that is provided in the "parameter\_1726= " Slider component.

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/lambda\_100**

<= code class=3D"svelte-114qcyq">

copy

curl -X POST http://localhost:7860/gradio\_api/call= /lambda\_100 -s -H "Content-Type:

application/json" -d '{ "data": [ true ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/lambda\_100/= $EVENT\_ID

Accepts 1 parameter:

[0]

b= oolean

Required

The input value that is provided in the "Show Ch= erry-pick Interface" Checkbox component.

**Returns 1 element**

string

The output value that appears in = the "Cherry-pick Interface" Textbox component.

**API name:**

**/load\_text\_from\_file\_100**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/load\_text\_from\_file\_100 -s -H "Content-Type:

application/json= " -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_100/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Text= to Generate from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Text to Generate" Textbox component.

**API name:**

**/generate\_multistyle\_speech**

copy

<= /div>

curl -X POST http://localhost:7860= /gradio\_api/call/generate\_multistyle\_speech -s -H "Content-Type:

applicatio= n/json" -d '{ "data": [ "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}},

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, {"path":"https://github.com/gradio-

app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

true ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/generate\_mu= ltistyle\_speech/$EVENT\_ID

**Accepts 302 parameters:**

<= span class=3D"code svelte-1yt946s" style=3D"margin-right: 10px;">[0]=

Required

The input value that is provided in the "Text = to Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[3]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[4]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[5]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[6]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[7]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[8]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[9]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[10]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[11]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[12]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[13]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[14]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[15]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[16]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[17]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[18]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[19]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[20]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[21]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[22]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[23]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[24]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[25]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[26]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[27]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[28]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[29]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[30]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[31]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[32]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[33]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[34]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[35]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[36]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[37]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[38]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[39]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[40]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[41]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[42]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[43]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[44]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[45]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[46]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[47]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[48]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[49]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[50]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[51]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[52]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[53]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[54]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[55]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[56]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[57]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[58]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[59]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[60]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[61]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[62]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[63]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[64]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[65]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[66]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[67]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[68]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[69]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[70]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[71]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[72]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[73]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[74]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[75]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[76]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[77]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[78]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[79]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[80]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[81]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[82]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[83]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[84]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[85]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[86]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[87]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[88]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[89]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[90]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[91]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[92]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[93]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[94]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[95]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[96]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[97]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[98]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[99]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[100]

strin= g

Required

The input value that is provided in the "Speech Type = Name" Textbox component.

[101]

any

Required

The input value that is provided in the "Regular Referen= ce Audio" Audio component. The FileData class is a subclass

of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and

metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

[102]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[103]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[104]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[105]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[106]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[107]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[108]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[109]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[110]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[111]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[112]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[113]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[114]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[115]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[116]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[117]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[118]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[119]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[120]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[121]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[122]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[123]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[124]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[125]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[126]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[127]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[128]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[129]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[130]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[131]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[132]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[133]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[134]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[135]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[136]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[137]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[138]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[139]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[140]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[141]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[142]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[143]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[144]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[145]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[146]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[147]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[148]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[149]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[150]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[151]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[152]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[153]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[154]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[155]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[156]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[157]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[158]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[159]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[160]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[161]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[162]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[163]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[164]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[165]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[166]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[167]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[168]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[169]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[170]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[171]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[172]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[173]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[174]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[175]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[176]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[177]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[178]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[179]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[180]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[181]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[182]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[183]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[184]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[185]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[186]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[187]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[188]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[189]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[190]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[191]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[192]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[193]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[194]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[195]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[196]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[197]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[198]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[199]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[200]

any

Required

T= he input value that is provided in the "Reference Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[201]

string

Required

[202]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[203]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[204]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[205]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[206]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[207]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[208]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[209]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[210]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[211]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[212]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[213]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[214]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[215]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[216]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[217]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[218]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[219]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[220]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[221]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[222]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[223]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[224]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[225]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[226]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[227]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[228]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[229]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[230]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[231]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[232]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[233]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[234]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[235]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[236]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[237]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[238]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[239]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[240]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[241]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[242]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[243]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[244]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[245]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[246]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[247]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[248]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[249]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[250]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[251]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[252]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[253]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[254]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[255]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[256]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[257]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[258]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[259]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[260]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[261]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[262]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[263]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[264]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[265]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[266]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[267]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[268]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[269]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[270]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[271]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[272]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[273]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[274]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[275]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[276]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[277]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[278]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[279]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[280]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[281]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[282]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[283]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[284]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[285]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[286]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[287]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[288]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[289]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[290]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[291]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[292]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[293]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[294]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[295]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[296]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[297]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[298]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[299]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[300]

strin= g

Required

The input value that is provided in the "Reference Te= xt" Textbox component.

[301]

boole= an

Required

The input value that is provided in the "Remove Si= lences" Checkbox component.

**Returns list of 102 elements**

[0]

T= he output value that appears in the "Synthesized Audio" Audio component.

[1]

string

The output value that appears in the "Reference Text (Regular)" = Textbox component.

[2]

string

The output value that appears in the "Reference Text" Textbox component.

[3]

string

The output value that appears in the "Reference Text" Textbox component.

[4]

string

The output value that appears in the "Reference Text" Textbox component.

[5]

string

The output value that appears in the "Reference Text" Textbox component.

[6]

string

The output value that appears in the "Reference Text" Textbox component.

[7]

string

The output value that appears in the "Reference Text" Textbox component.

[8]

string

The output value that appears in the "Reference Text" Textbox component.

[9]

string

The output value that appears in the "Reference Text" Textbox component.

[10]

string

The output value that appears in the "Reference Text" Textbox component.

[11]

string

The output value that appears in the "Reference Text" Textbox component.

[12]

string

The output value that appears in the "Reference Text" Textbox component.

[13]

string

The output value that appears in the "Reference Text" Textbox component.

[14]

string

The output value that appears in the "Reference Text" Textbox component.

[15]

string

The output value that appears in the "Reference Text" Textbox component.

[16]

string

The output value that appears in the "Reference Text" Textbox component.

[17]

string

The output value that appears in the "Reference Text" Textbox component.

[18]

string

The output value that appears in the "Reference Text" Textbox component.

[19]

string

The output value that appears in the "Reference Text" Textbox component.

[20]

string

The output value that appears in the "Reference Text" Textbox component.

[21]

string

The output value that appears in the "Reference Text" Textbox component.

[22]

string

The output value that appears in the "Reference Text" Textbox component.

[23]

string

The output value that appears in the "Reference Text" Textbox component.

[24]

string

The output value that appears in the "Reference Text" Textbox component.

[25]

string

The output value that appears in the "Reference Text" Textbox component.

[26]

string

The output value that appears in the "Reference Text" Textbox component.

[27]

string

The output value that appears in the "Reference Text" Textbox component.

[28]

string

The output value that appears in the "Reference Text" Textbox component.

[29]

string

The output value that appears in the "Reference Text" Textbox component.

[30]

string

The output value that appears in the "Reference Text" Textbox component.

[31]

string

The output value that appears in the "Reference Text" Textbox component.

[32]

string

The output value that appears in the "Reference Text" Textbox component.

[33]

string

The output value that appears in the "Reference Text" Textbox component.

[34]

string

The output value that appears in the "Reference Text" Textbox component.

[35]

string

The output value that appears in the "Reference Text" Textbox component.

[36]

string

The output value that appears in the "Reference Text" Textbox component.

[37]

string

The output value that appears in the "Reference Text" Textbox component.

[38]

string

The output value that appears in the "Reference Text" Textbox component.

[39]

string

The output value that appears in the "Reference Text" Textbox component.

[40]

string

The output value that appears in the "Reference Text" Textbox component.

[41]

string

The output value that appears in the "Reference Text" Textbox component.

[42]

string

The output value that appears in the "Reference Text" Textbox component.

[43]

string

The output value that appears in the "Reference Text" Textbox component.

[44]

string

The output value that appears in the "Reference Text" Textbox component.

[45]

string

The output value that appears in the "Reference Text" Textbox component.

[46]

string

The output value that appears in the "Reference Text" Textbox component.

[47]

string

The output value that appears in the "Reference Text" Textbox component.

[48]

string

The output value that appears in the "Reference Text" Textbox component.

[49]

string

The output value that appears in the "Reference Text" Textbox component.

[50]

string

The output value that appears in the "Reference Text" Textbox component.

[51]

string

The output value that appears in the "Reference Text" Textbox component.

[52]

string

The output value that appears in the "Reference Text" Textbox component.

[53]

string

The output value that appears in the "Reference Text" Textbox component.

[54]

string

The output value that appears in the "Reference Text" Textbox component.

[55]

string

The output value that appears in the "Reference Text" Textbox component.

[56]

string

The output value that appears in the "Reference Text" Textbox component.

[57]

string

The output value that appears in the "Reference Text" Textbox component.

[58]

string

The output value that appears in the "Reference Text" Textbox component.

[59]

string

The output value that appears in the "Reference Text" Textbox component.

[60]

string

The output value that appears in the "Reference Text" Textbox component.

[61]

string

The output value that appears in the "Reference Text" Textbox component.

[62]

string

The output value that appears in the "Reference Text" Textbox component.

[63]

string

The output value that appears in the "Reference Text" Textbox component.

[64]

string

The output value that appears in the "Reference Text" Textbox component.

[65]

string

The output value that appears in the "Reference Text" Textbox component.

[66]

string

The output value that appears in the "Reference Text" Textbox component.

[67]

string

The output value that appears in the "Reference Text" Textbox component.

[68]

string

The output value that appears in the "Reference Text" Textbox component.

[69]

string

The output value that appears in the "Reference Text" Textbox component.

[70]

string

The output value that appears in the "Reference Text" Textbox component.

[71]

string

The output value that appears in the "Reference Text" Textbox component.

[72]

string

The output value that appears in the "Reference Text" Textbox component.

[73]

string

The output value that appears in the "Reference Text" Textbox component.

[74]

string

The output value that appears in the "Reference Text" Textbox component.

[75]

string

The output value that appears in the "Reference Text" Textbox component.

[76]

string

The output value that appears in the "Reference Text" Textbox component.

[77]

string

The output value that appears in the "Reference Text" Textbox component.

[78]

string

The output value that appears in the "Reference Text" Textbox component.

[79]

string

The output value that appears in the "Reference Text" Textbox component.

[80]

string

The output value that appears in the "Reference Text" Textbox component.

[81]

string

The output value that appears in the "Reference Text" Textbox component.

[82]

string

The output value that appears in the "Reference Text" Textbox component.

[83]

string

The output value that appears in the "Reference Text" Textbox component.

[84]

string

The output value that appears in the "Reference Text" Textbox component.

[85]

string

The output value that appears in the "Reference Text" Textbox component.

[86]

string

The output value that appears in the "Reference Text" Textbox component.

[87]

string

The output value that appears in the "Reference Text" Textbox component.

[88]

string

The output value that appears in the "Reference Text" Textbox component.

[89]

string

The output value that appears in the "Reference Text" Textbox component.

[90]

string

The output value that appears in the "Reference Text" Textbox component.

[91]

string

The output value that appears in the "Reference Text" Textbox component.

[92]

string

The output value that appears in the "Reference Text" Textbox component.

[93]

string

The output value that appears in the "Reference Text" Textbox component.

[94]

string

The output value that appears in the "Reference Text" Textbox component.

[95]

string

The output value that appears in the "Reference Text" Textbox component.

[96]

string

The output value that appears in the "Reference Text" Textbox component.

[97]

string

The output value that appears in the "Reference Text" Textbox component.

[98]

string

The output value that appears in the "Reference Text" Textbox component.

[99]

string

The output value that appears in the "Reference Text" Textbox component.

[100]

string

The output value that appears in the "Reference Text" Textbox component.

[101]

string

The output value that appears in the "Cherry-pick Interface" Tex= tbox component.

**API name:**

**/validate\_speech\_types**

copy

=

curl -X POST http://localhost:7860/grad= io\_api/call/validate\_speech\_types -s -H "Content-Type:

application/json" -d= '{ "data": [ "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!",

"Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!", "Hello!!" ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/validate\_sp= eech\_types/$EVENT\_ID

**Accepts 102 parameters:**

<= span class=3D"code svelte-1yt946s" style=3D"margin-right: 10px;">[0]=

Required

The input value that is provided in the "Text = to Generate" Textbox component.

[1]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[2]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[3]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[4]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[5]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[6]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[7]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[8]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[9]

string<= /span>

Required

The input value that is provided in the "Speech Type Na= me" Textbox component.

[10]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[11]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[12]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[13]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[14]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[15]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[16]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[17]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[18]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[19]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[20]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[21]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[22]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[23]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[24]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[25]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[26]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[27]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[28]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[29]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[30]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[31]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[32]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[33]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[34]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[35]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[36]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[37]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[38]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[39]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[40]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[41]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[42]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[43]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[44]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[45]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[46]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[47]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[48]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[49]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[50]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[51]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[52]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[53]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[54]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[55]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[56]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[57]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[58]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[59]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[60]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[61]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[62]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[63]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[64]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[65]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[66]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[67]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[68]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[69]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[70]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[71]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[72]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[73]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[74]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[75]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[76]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[77]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[78]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[79]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[80]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[81]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[82]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[83]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[84]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[85]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[86]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[87]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[88]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[89]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[90]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[91]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[92]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[93]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[94]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[95]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[96]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[97]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[98]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[99]

string=

Required

The input value that is provided in the "Speech Type N= ame" Textbox component.

[100]

strin= g

Required

The input value that is provided in the "Speech Type = Name" Textbox component.

[101]

strin= g

Required

The input value that is provided in the "Speech Type = Name" Textbox component.

**Returns 1 element**

**API name:**

**/lambda\_101**

<= code class=3D"svelte-114qcyq">

copy

curl -X POST http://localhost:7860/gradio\_api/call= /lambda\_101 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_101/= $EVENT\_ID

Accepts 0 parameters:

**Returns 1 element**

**API name:**

**/load\_chat\_model**

copy

=

curl -X POST http://localhost:7860/gradio\_api= /call/load\_chat\_model -s -H "Content-Type:

application/json" -d '{ "data": [ "Qwen/Qwen2.5-3B-Instruct" ]}' \ | awk -F'"' '{ print $4}' \ | read

EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/load\_chat\_m= odel/$EVENT\_ID

**Accepts 1 parameter:**

[0]

s= tring

Required

The input value that is provided in the "Chat Mod= el Name" Dropdown component.

**Returns 1 element**

**API name:**

**/load\_text\_from\_file\_101**

copy

curl -X POST http://localhost:7860/gr= adio\_api/call/load\_text\_from\_file\_101 -s -H "Content-Type:

application/json= " -d '{ "data": [ {"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=

es/sample\_file.pdf","meta":{"\_type":"gradio.FileData"}} ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID;

curl -N http://localhost:7860/gradio\_api/call/load\_text\_f= rom\_file\_101/$EVENT\_ID

**Accepts 1 parameter:**

[0]

a= ny

Required

The input value that is provided in the "Load Refe= rence Text from File (.txt)" File component. The FileData class is a

subclass of the GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data

and metadata when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized

server URL pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload.

mime\_type: The MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used

internally (should not be changed).

=

**Returns 1 element**

string

The output value that appears in = the "Reference Text" Textbox component.

**API name:**

**/process\_audio\_input**

**Handle audio or text input from user**

=

=

copy

curl = -X POST http://localhost:7860/gradio\_api/call/process\_audio\_input -s -H "Co= ntent-Type:

application/json" -d '{ "data": [ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=

"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],

{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil= es/audio\_sample.wav","meta":

{"\_type":"gradio.FileData"}}, "Hello!!" ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/process\_aud= io\_input/$EVENT\_ID

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Speak you= r message" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_text\_response**

**Generate text response from AI**

copy

curl -X = POST http://localhost:7860/gradio\_api/call/generate\_text\_response -s -H "Co= ntent-Type:

application/json" -d '{ "data": [ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=

"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}], "Hello!!" ]}' \ |

awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/generate\_te=

xt\_response/$EVENT\_ID

**Accepts 2 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

s= tring

Required

The input value that is provided in the "System P= rompt" Textbox component.

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_audio\_response**

**Generate TTS audio for AI response**

copy

**cur= l -X POST http://localhost:7860/gradio\_api/call/generate\_audio\_response -s = -H**

**"Content-Type: application/json" -d '{ "data":**

**[ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=**

**"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],**

**{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=**

**es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!", true, true, 3 ]}' \ |**

**awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N**

**http://localhost:7860/gradio\_api/call/generate\_au= dio\_response/$EVENT\_ID**

**Accepts 6 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Reference= Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

[3]

boolean=

Required

The input value that is provided in the "Remove Silenc= es" Checkbox component.

[4]

boolean=

Required

The input value that is provided in the "Randomize See= d" Checkbox component.

[5]

any

Required

The input value that is provided in the "parameter\_1764" N= umber component.

**Returns list of 3 elements**

[0]

T= he output value that appears in the "value\_1773" Audio component.

[1]

string

The output value that appears in the "Reference Text" Textbox component.

[2]

The output value that appears in the "value\_1764" Number component.

**API name:**

**/lambda\_1\_1**

<= code class=3D"svelte-114qcyq">

copy

curl -X POST http://localhost:7860/gradio\_api/call= /lambda\_1\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_1\_1/= $EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Speak your message" Audio component.

[1]

string

The output value that appears in the "Type your message" Textbox component.

**API name:**

**/process\_audio\_input\_1**

**Handle audio or text input from user**

copy

**cur= l -X POST http://localhost:7860/gradio\_api/call/process\_audio\_input\_1 -s -H= "Content-**

**Type: application/json" -d '{ "data":**

**[ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=**

**"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],**

**{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=**

**es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!" ]}' \ | awk -F'"'**

**'{ print $4}' \ | read EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/process\_aud=**

**io\_input\_1/$EVENT\_ID**

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Speak you= r message" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_text\_response\_1**

**Generate text response from AI**

=

<= div class=3D"copy svelte-114qcyq">

copy

curl -= X POST http://localhost:7860/gradio\_api/call/generate\_text\_response\_1 -s -H= "Content-Type:

application/json" -d '{ "data": [ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=

"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}], "Hello!!" ]}' \ |

awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/generate\_te=

xt\_response\_1/$EVENT\_ID

**Accepts 2 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

s= tring

Required

The input value that is provided in the "System P= rompt" Textbox component.

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_audio\_response\_1**

**Generate TTS audio for AI response <= /h3>**

copy

**c= url -X POST http://localhost:7860/gradio\_api/call/generate\_audio\_response\_1= -s -H**

**"Content-Type: application/json" -d '{ "data":**

**[ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=**

**"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],**

**{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=**

**es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!", true, true, 3 ]}' \ |**

**awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N**

**http://localhost:7860/gradio\_api/call/generate\_au= dio\_response\_1/$EVENT\_ID**

**Accepts 6 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Reference= Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

[3]

boolean=

Required

The input value that is provided in the "Remove Silenc= es" Checkbox component.

[4]

boolean=

Required

The input value that is provided in the "Randomize See= d" Checkbox component.

[5]

any

Required

The input value that is provided in the "parameter\_1764" N= umber component.

**Returns list of 3 elements**

[0]

T= he output value that appears in the "value\_1773" Audio component.

[1]

string

The output value that appears in the "Reference Text" Textbox component.

[2]

The output value that appears in the "value\_1764" Number component.

**API name:**

**/lambda\_2\_1**

<= code class=3D"svelte-114qcyq">

copy

curl -X POST http://localhost:7860/gradio\_api/call= /lambda\_2\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_2\_1/= $EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Speak your message" Audio component.

[1]

string

The output value that appears in the "Type your message" Textbox component.

**API name:**

**/process\_audio\_input\_2**

**Handle audio or text input from user**

copy

**cur= l -X POST http://localhost:7860/gradio\_api/call/process\_audio\_input\_2 -s -H= "Content-**

**Type: application/json" -d '{ "data":**

**[ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=**

**"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],**

**{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=**

**es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!" ]}' \ | awk -F'"'**

**'{ print $4}' \ | read EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/process\_aud=**

**io\_input\_2/$EVENT\_ID**

**Accepts 3 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Speak you= r message" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_text\_response\_2**

**Generate text response from AI**

=

<= div class=3D"copy svelte-114qcyq">

copy

curl -= X POST http://localhost:7860/gradio\_api/call/generate\_text\_response\_2 -s -H= "Content-Type:

application/json" -d '{ "data": [ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=

"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}], "Hello!!" ]}' \ |

awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N http://localhost:7860/gradio\_api/call/generate\_te=

xt\_response\_2/$EVENT\_ID

**Accepts 2 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

s= tring

Required

The input value that is provided in the "System P= rompt" Textbox component.

**Returns 1 element**

The output value that appears in the "C= onversation" Chatbot component.

**API name:**

**/generate\_audio\_response\_2**

**Generate TTS audio for AI response <= /h3>**

copy

**c= url -X POST http://localhost:7860/gradio\_api/call/generate\_audio\_response\_2= -s -H**

**"Content-Type: application/json" -d '{ "data":**

**[ [{"role":"user","metadata":None,"content":"Hello!","options":None},{=**

**"role":"assistant","metadata":None,"content":"How can I help you?","options= ":None}],**

**{"path":"https://github.com/gradio-app/gradio/raw/main/test/test\_fil=**

**es/audio\_sample.wav","meta":{"\_type":"gradio.FileData"}}, "Hello!!", true, true, 3 ]}' \ |**

**awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N**

**http://localhost:7860/gradio\_api/call/generate\_au= dio\_response\_2/$EVENT\_ID**

**Accepts 6 parameters:**

[0] <= span class=3D"code highlight svelte-1yt946s" style=3D"margin-right: 10px;">= any

Required

The input value that is provided in the "Conversat= ion" Chatbot component. null

[1]

a= ny

Required

The input value that is provided in the "Reference= Audio" Audio component. The FileData class is a subclass of the

GradioModel class th= at represents a file object within a Gradio interface. It is used to store = file data and metadata

when a file is uploaded. Attributes: path: The server file path where the file is stored. url: The normalized server URL

pointing to the file. size: The size of the file in bytes. orig\_name: The original filename before upload. mime\_type: The

MIME type of the file. is\_stream: Indicates whether the file is a stream. meta: Additional metadata used internally

(should not be changed).

=

[2]

string

Required

[3]

boolean=

Required

The input value that is provided in the "Remove Silenc= es" Checkbox component.

[4]

boolean=

Required

The input value that is provided in the "Randomize See= d" Checkbox component.

[5]

any

Required

The input value that is provided in the "parameter\_1764" N= umber component.

**Returns list of 3 elements**

[0]

T= he output value that appears in the "value\_1773" Audio component.

[1]

string

The output value that appears in the "Reference Text" Textbox component.

[2]

The output value that appears in the "value\_1764" Number component.

**API name:**

**/lambda\_3\_1**

<= code class=3D"svelte-114qcyq">

copy

curl -X POST http://localhost:7860/gradio\_api/call= /lambda\_3\_1 -s -H "Content-Type:

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/lambda\_3\_1/= $EVENT\_ID

Accepts 0 parameters:

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Speak your message" Audio component.

[1]

string

The output value that appears in the "Type your message" Textbox component.

**API name:**

**/clear\_conversation**

**Reset the conversation**

= copy

curl -X POST http://= localhost:7860/gradio\_api/call/clear\_conversation -s -H "Content-Type: appl=

ication/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/clear\_conve= rsation/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Conversation" Chatbot component.

[1]

The output value that appears in the "value\_1773" Audio component.

**API name:**

**/clear\_conversation\_1**

**Reset the conversation**

copy

curl -X POST http:= //localhost:7860/gradio\_api/call/clear\_conversation\_1 -s -H "Content-Type: =

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/clear\_conve= rsation\_1/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Conversation" Chatbot component.

[1]

The output value that appears in the "value\_1773" Audio component.

**API name:**

**/clear\_conversation\_2**

**Reset the conversation**

copy

curl -X POST http:= //localhost:7860/gradio\_api/call/clear\_conversation\_2 -s -H "Content-Type: =

application/json" -d '{ "data": [ ]}' \ | awk -F'"' '{ print $4}' \ | read EVENT\_ID; curl -N

http://localhost:7860/gradio\_api/call/clear\_conve= rsation\_2/$EVENT\_ID

**Accepts 0 parameters:**

**Returns list of 2 elements**

[0]

T= he output value that appears in the "Conversation" Chatbot component.

[1]

The output value that appears in the "value\_1773" Audio component.

=09 =09



=

------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/png

Content-Transfer-Encoding: base64 Content-Location: chrome-

extension://difoiogjjojoaoomphldepapgpbgkhkb/assets/logo-O35E636P.png

iVBORw0KGgoAAAANSUhEUgAAAMAAAADACAYAAABS3GwHAAAACXBIWXMAACE4AAAhOAFFljFgA

AAA AXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAHFHSURBVHgB7b19kJ3pVR94zvPe1oyN8bTW2IuD

ja62QlXYFDUSWyHkj2VaVLYWqkisAYPHH6CWv9YORSQZKGzAVvcQ1k62QBIVggOO1YOxGcDLyFl2

YzYU6iFmoTaVlabIrlP5Y6dVSdbgj+22PR7NqO99Tp7nOZ/v7Zb6SnN7pjX0M9Pq2/e+9/08H7/z O+c5D8D

+2B/7Y3/sj/2xP/bH/tgf+2N/7I/9sT/2x/7YH/tjf+yP/bE/9sf+2B/7Y3/sj/2xP/bH /tgf++PFMBD+kgy6eGT

+a5tzw7wJR9JocB+N5g5CHpRPBpRvDBDhwBpuDtZGMFg7uPToGrwIx58s

fHaYunRkRDifgYaQAHJ5P0GiEcCXIeU1SGnj2Ke/ZRX+kowXtQJcv/hXF8a5e10R9AUYD46UH0Ca IxoVwS/CT

+M5KK8R81z5u7wun9dtyusNGM2tlt+fGm8eWD14/vwa3IXjysKT82N49gQlXMhED5TH

PT9GQiqPnbAKf/ld/gNMTREIkcpfkLF8gHh1DLBKmD71vZ8+vAov0vGiUwB6bDj/zFNwajxKp5GK

pR/PIbhgi5DP2d/8ewC5KEVThHFHlA8gjLoiG3NYlQRGg9VEg5WX//IHH4G7YFxZ+OxCEd6z5fE+

UAQdRbiLkFdxZ+F3JeDPquDzD1JO5Z+iJ6wI7b21okSr6QAsf++lw2vwIhovGgW4/pvfOCzO/Gwe

dYtFkKkJfi7CPBIFyAMqgl+eKv+uf7P1H5AqCalCjFkRioKwArTXB+p31mhzsHTw4k/vSUW4svB/

LxbBPVsEetgEuVr79n/5QxWgCTtWK98Evlh6yqwM2Kw/ACsIb0fhPZTtVqgowoMvEkW46xWgWfwb

T58qlr38dAebVa8Qh1SwO2oC3YRblKK9L9ZePUQfAm15jeM5YiUZ1O3XYPMlxw4++u412AOjWvzy

IM8WaX2gCn0VVDCIwwqgAt9+mjIEy28eQJQG+p9XJRmXv+L25Y+lB/+3Q8twl4+7WgGe+u35Iykf

eAyoO1SEHklhjVh+9gRFYO33gLBsk8dzzfI3jyDxQBHqAn3K++V3Zg/RPEFRqva7KoAIPytFhUzj

A0sHP/m2F1QInviu/+dcEe7ToEJdjHyDMCLEKMJbcH6DNNR0oFxGFegS/FJqAl+8QJVpgUaAuq17

g7pfgVMlUGZvQrg2nsvH3ngXe4O7VgGe+eR/cSqP0tkilAfdwrOwkghvC3Qbtu/DHGw4v/w9YqFX

YZdYQSz9Ad5+bAoS9lm8QR7w90aDq4iDBw9eeuMaPI+jWP1h8W0XiyAvgEKVVKS9CGbzACzEYDEA gHsDwf

+yjX2eMXiO8DnHA/WbVXH4/YnY4syb/pdD5+EuHHelAlz/nfki+N1ZEkveBL5i/Ykgl8Sy Y8P

+bs01HlBsX9+nZuUH+v2CIxTuiPcYM2skylKVBMkU48BayunYwU8/uAbPw/hsEf5nM10uLw+x tfegtgkkuYA2jN

+EGEIsABYAF3NusMe/L7GCeARTIHTvUJVg3PaWSPa3/OZ//tq7DhIluMsGC/9g

qVGaTajnoqUmFshOhL1Zf1UEZnn4NfI2c+gQZ0Dt+7l5EMxR+Pm98sgVYonwZ4Nch3I+cHn9e/7F

EHZ5VMt/g4rwIx7iwLXasAJlVDgBLKAVIfdAuL5silAhTdWJFgwbtp8QdP7h7Wz/hOot2Hg25UlN 2ZY

+9nf/01m4y8ZdpQDX/+ci/JTONkGlOYMnJJi9sjqgAp/n+gFtiwu6Jugk3kHiAJDvoFp6zKpc

HccVokxl/wTbQaLqDcbdMF+/5w/XF/5kCLs0quUvon65CNswBq8c+FITfGgCCyao9hpZ2N0bOPXZ

vqPsEAQGqCpQjREM7gRPAyGYZu9Sz+PsI8fvLiW4ayBQs/w0WGpCzMksoTAtKC1PRpga4/6d5yfG

9shUp9OcLQ5oCbIChcq+KSpO+RuyBMBjUa5xX7nAYZMqxVp34CVHD64e3YAZjs+K5acm/NiHLe23

8fwS0Cqe38ruQKowByZgjsKe4A0CBKp6M5bguh9PeKDssUY687ZLf+WuiAnuCg9Q2R6ibqnBlWDp

s8KTUfht9GUV7I4tP7+HHAAPXPjra2pskQu2QpvyXYVMFgjX72dVshYAk9GlY45FSmwyHF2nx2DG

41mii0XoDin8MLYGRMB7wh+CXocvHNiaoCqc8ffVI0xa+0aDKqySz5vXwW2EvykR/eKvvf5zC3AX jD2vANcfmx

+m3D1m+F6EcgK+NMwvHsHgkMYIyNaflAViBqhr7wkc4kA4d9DLGmfh/0kYn7HvPwbJ qjSmMBkf+PO/

+W9Ow4zGn33XZwusoAUCpiwj/AHF8ZrMUiuOKqjVSpMFyhr4VtwObM2pBr3ZFKXt

k0DjBdLjhYBYlaNSo/UYBPY92bbEUHTx4vEn52GPjz2vAHmzsD3UDTUYVevL1r9zqx8+89hg4ApC

Cou6IOBzGLdHhzGkya8m5AybnPqM9GjuWAm4ZIIqsd6wOHRnP/edV4bwHEcNegvbssQBK4IyMIbB

g7Vn1gd6WB7U0otisKUnteRS7tBRk3hRAgmMS3CLLcB12ANGj7Z91s9bDI7gnsk8x/DZ9JI9Hw/s

aQV46rdfeaRY2kUU4cbckVrfnjdQ7J8ZzqCwOSVBRigJrlAOYUGvK0MMlv1HAmMQbwHBYygVSiBK

UrxHgQSInnWl+SIlF+E5jiJq5yBYZxU0hSEmnEkpzhCkgsMlVRT+HprHaJAIyWDTNpAqehiIyiA0

ayixACmoYwUdYz79K3scCu1pBUjj9JgnouZqwVqw2HNi7TUGUIjUKcdfcwPsKcZ9+pIpzzkOeBX2

jCSf0Kx/Z96CFYi/l8fuAUCywk2RRvU2BlYFOKlUBOSB//Cdf7YAdzhqbU/Z7+tq1jZAjyZg7AVQ

eX7m7vn4pAoToE9DTxkd+2eFSO1IIR5of0ZBx15tUDtes/zMn2TwbXNkpeoLTMBFeXt37FkFeOrj

37hYA0qhH51zNwVgq6xeoZU1mGJYRhjNmrO1p2j90Xj8OYVAAfIIdLJ8QIe6jSmfKgF0oe5GisvA

AsI7FwDCswp5VOAb584BqBSvKRvkCthKH9xrcA4gCXSSxJkpkAbSJJ9p7kA8g8EtsFIIjJ5C92GF c

+YtjJV64Nwe9gJ7VgESDk4wlOnImZmWjCK0wjbH/C2AlZjAYYpY+zHXAGmAq9neplhjVo7G+Tc8

31nc4AVwnZQ9dKFITr5bMD+I2xc+Hj1jWoWTFp68Ay9QC9yKMA2t3EDxNTkcyhgZG98OcAKPl19j

Aqn/CYLavk9Avj2BZHbVsgfGCSgcCzBUkKLSrqKk7XPdpqIs3LNeYE8qwPXfHA5LBuYBgTgs0COh

NEfC2IwHhvUDI8S4PNT5UwyOJRZQWGPUJgexpLkDCYBBEmOEGuQSwyhTyBJj1Ac9bhazZ3VBgkqG

GN0dwYATfRzPAq8WXILRrRWdAeZ4AMxYn4WTq0VbtoyhDZrFB4YvLMiaQIuUKe9T4gkKFaM1ZmjK

b5AMOEEmVaQL/9NDX1iAPTj2pALkzXyW2R2p2nSoggGykMQESIbNJwSUoYp+Tk6ZzpnlF+yPIfCN

8Ad6yhRYpooX2OKbcAXo07eUBZ8vPLkwPSXYmB+gxZ5CMaOjggUQhHIigA2Bb9vei98QreCtBask

iuLKw98jU2CqrxVy1YpSgVdq9ckn2XBJhlKi6lVAzgtyXoA9OPYmBBoPFnqMjBWhaea3D0N4DkBL

eHGmdySwaeTwyet+mLokSXQx7JFA12DSnMImqp9bGTR/twm/4140yOCZVwkI0T979tmvLE57+Rmq

sFRLipNY3oULfMJKv4SBFUSYHLRsL2JfUVpcgFzpCYEhkphBK0k98JbYAuNx+yzSWNmktk2SYyax

A3gC9uDYcwrw1MVvOVIEbmiwZTSIWF+5fw+KDYvPSZBapzLylMYquFLkFqhTy+g6/DEYFLj9GAy3

JJl9ForINLhkWcix2rJdDdOP1XIWCViY9h6UjV+nsGcsTA5bcsX0HLCqVe57CmChQ4cnTZlasivE

CgDgFGeP38eYMe6VPtt36hbJZotpkA687xg4C0Xb5hwPP/jQ54awx8aeUoA6ib081bOMrwc0weAI

pJkzi25WuQq9WPucI+05B1rV6Rhfg14ubpPyBYwKRaZwXAyHxgp1oCyI0ouWbRVBgol6GlaC8k5K

D0x7HzKmBRV0DbADFiedoBK9j8cB6HGJnE/uZYj7eQKFUpkhk3icZHQr5wz610U6oUYL6IwdUqXw

Mon2mVC0iQanYI8NhD0wrn9sWCAP1jLnB3S6IurUReH3feoiT26x2V6WoPL5AJb8akpxAPozuboe

rGJvMhcrO3uVnjYrrByPXb0xIuIJ+lMFe1AEDK5wdrjbPPytq0fXbnUv/s+Ff3+kiM8VFnaZrdWw

tLAzfryWvc3bB8H9DLFZ+YnzBBAPhQEymSJtKYiLJRZWRgEMlyBcu3oFLrwDLceW86FrRcGWfvo3 v

+ER2APjBfUAle15+mOvvZzHeLlY6oUg5GbBGbtr8OsMjwS3zu40RekCz8+FbqD8f/0R4UctbWiK

dkA8TdcTfupRrAMTBCCnETUTa1y6/VCjF806SwlyHh9Y2OmeFGJpqMxNDtabBZ8wKBYLHkwGwqYc

JrB9dqoFtrQ1eJbtKU6E75c7R6/Rrs9iBOiXXoDRrGhBsWCx8v+w/HHxf3zjl578+Yc+vwgv8HjB

FODpj7/6VB6Nr1Dt2aN1OFZU1hFncGuyqwsMjCiDCrQExqYQ2SfHUIsVOlUGzdoq/8/BrfD/MquL

IHD8Ghsw5hcgrS4doG/hQ0cFxenQYIQGkMbPH9rpvmTKR8ziAkwEtdDqbrhQDRX78w+pNQcudQhw

JaNnhEljl62KIbEFCmukNUPYE2w5N4NUOUzI0W08JuAf8SyqvBJT5KLo6eI/eNMXLy49tD6EF2g8

7wpQuzg8/YlXnyPqzhdBmxd40wRWKE+elB5wPAu5VHuKUItgE8YaHhFgNIhjRWyetRW832qGitCT wZ6uh/

+51KGjFtShQx5AVJk0TJ3DBBMItTAUkkXcfGqa281QJxa8cesS4ont5DFHyMyi1OCIEpJX aso

+VfilHGK1KNJKeX0NRHC5YRbxZBkAjTHAYwSDQxyXREYqgWWlx2r5tQwDlTli1SCNTRh91+B8 MaX8h0tv

+vwReAHG86oAVfivP/3s5WJ1TxvGtvLmOW1XgjFYFetvjIxNSxQhzSGh5UFz4PSl1Nm8

S7P0c8LsdDIDbM5Lm43zT8a8RLbEqUUT/sCjYwySKeLxLAK507B6fhT+Xo+ttT3oJcsE8bjGFgH0

cL8Io06JzHDmod/75mMPfeqbT77pn3/z4XJaK5wb8GpT82oCvbRvECsBaUa656WkyK49ZnZYCJoc

5FPSXETjThlCiVKXDYZFFK8sv3n9BDzPYwDP02jCf70If+6OKJ2o0KSWN1TIg+PwfmR6RLjbNhOz

sVqHB7XgFFqjqBK1YjctZeZpjhXv8yyv7svFE10o371aGIwNNWcwKvFztQ1yd0Zyo8YieiMge/JY

PqmhhfIJLIC83ZjkdQfSTWGwtvOdSiSiIXuSANrm9UZF00C5eSApUCNXIugFyNWoP/HQ/9rv3rCJ

4zOJcDGHak/PaWiVZ4h1ULFNcxZaQo1kNr4dr30ohgFj/EC2RcyYE+8bxisfeNOXhg9/4hXL8DyN 500Brl

+/frkEpUeClUbF+57QMuanX3PTS3x1LMzC88NYpkZKbT8Hv/Z9hkhe9cnf5XjgWuq6Y6/4 9PeuwR4bLiwq9Gjcu

+J5FkaGF5Ga5aZY2NuHKAQW2HN18lgnLx3eWDn+H9eqFdbyCAAPpusbFuhy

pzlhgvjYjeZtfJRBQFYGFHegcQSfs7yD4TpZ+PXv8tHSB968vvbwxw8+As/DeF4gUMX8RSjvj2XM

jMG9xNlmc40HE9MTtYObfDdHpsh6/sgk99ALiGt4tIQZfeZX+97V9NLNowf3oPAzzEjGHvGMqyoa

qeF0DjgBdEZXPwdQR5jBZcVxHKiWjbctx3CsLzVA4JNoxlLu4XMQJmCPHVMrVpsWmUC78PPb5pUA

/P2WN5A2jm2feeWnn6eYYNcVoLI95WmdYu6+ozB1USo4Fe+7YFdIFIXcpjWKB5AaHfQAunmENvE9

tish4kyw8PgSQ8ytvOJ//ztHD156cKaT1mc1RilLJjb1BVFrcASOZO7UZu1NekE5OJRp1pqUksXt FQBpw2hLS

+oxfIGYQAvHkew2K2kSD9NGyCYLWYCiIPIH1yLpscE9FQSaF7B77L2Lu88O7aoC1Ia1 PJnd+vD0hX0s7cqlPp

+0zr42uSJRklFgZ0ad8/TiFbSWp1VqeuOrWOPTYoCK+TOl5Vf8/vedhD0+ NNjuzcRCF+j6yzLPiWuOYnKLgoDx/F

+U/dFwu+OV4HXNA+gUK1p7+Q05lr0/JgUwfCpWHlIFn+EU SiyhTFWw/ErjArCHqu

+z0ksZxyHYpIuwy2NXY4A87s4W7D1PnlndUnUpAW+v76ZYeQwVndVrTLI9 NkFlS

+LLKzolzhisj4je86pPH1+52bnebP7uM/Wfe7f/zjO91/fKOzfZuI17N46uHt7B86RQegw6

VxfAA16BLDFAFcyuQWejUslmbbXteTGMbT1A8RDXzMIzawQQqkqFlYLQXU7ikER6fNDzJfEGvPCA

BbmiKsCCL6yUUa2gwbJAJ2RuDfGB9/7wFxc/9LFvWIFdGrumAE/9+jctInUnWiuSGuhSBxSnFEpZ A1ggW

+vtBzbzyvH/AFIWlijPCRM08JKHmDkeeZDLBW6NSl3DETz4qn/54NWbnevn/+aVUznDOauZ

0QddHnCVIbohbQb7WU6m85Alc66h2AMVv4IzMPK5WL4xPrNcfi3tdO+4dMCoSw+G7dykA7QzNyaI

yr0ThdcWCOP8xeOfG5689Oq1eLzCVl1FTWSJ6OXIIrVzShTmCgCXgbeImEC9EXgQLZCILPiVxBu0

76LcYQ2UQdMCQcGZhSrw+dzS4vqlpZWDuwJZdw0CJRicbVCG+/LgZE29tSs3zn+gHR68ilML3LT/

jwXLBotQef76k4XXD8Vsa4nSsYO3EP6/+I4rZ4sAnO+5/V75gQo/eNIH0XIDlk1F6GHlUBFqlZF1

KaKd7luulZOmND5JxfC+KKYGrMqnU68WpxcLeE6gHWFzm+CyW42KnaMHUnglpc7GKAGAeQewoNgE

2rF8XymYPQrKKkrCMYLeKwKFRuX3/NMj2LUiul1RgOsf+6sLUEua88CbVoU5uxLMgpUnkJc1SEKK

W5YEarOVOSuUGg8Czx9yAqHEuSjQaronHb1Zw9r1hSvzn/8bVy5SnbStD5yTMhz4hamH0AJO494h

B2qPhYxgSxGYFo4ZhegCcquRq6g1eU2k0xVNidKEYnqwCs7C6LVUPfFMtJYqlP0+MHnMd1ePkEp2

GLA3AYcnvwP6vsUbKSST75MymxhZIbCkGcg5ZckNgFO1EiNrQpGL/ppnMzqpecPTxQvsSo+hXYFA eUwn

+ty9YXuI8EbjgByaTWF4X5iftqqLQZ9WyhCo1NHgaqbB1cr9FyUZFqtflC/90iv+xffftDHV

esH7zz5Nj5Ubf79MCYRYzdgeVmLs2qxpspLgPiZXLCtJI+bpg1CiPlwKwd9OI5HicBE4+aHAt+t5

AJ9H0uQUmPCSZa6Vt4cmYEXQtqUXc1sPDBZ4u2RBrHkG3a9MtiT1fsCKiYrbQXMEDqH0NcZzBIE6

niiW/AF4rQnwkYom3/e1XGQK4ALMeMxcAdYvDmt9z2KDPmLNlb5E6pc1+OSUOOHFk13WoSFMbAFh lMq

+VsttO3Pwt9/Vgzfrxx+bvxXFWaz+kRvjIvwJh1L3IsLkc2XLZ6SshTwheXBskUDm0YIGmwpN

BG8zNid0AdD94HQQiDLEWhvmyaV2BoJwa9BLEIRcWBrayrq07RIu1I5tNQEWj3tj9MyFbu6lp1s/

o3r2yT1LE/pyXq1VXMDyep3tvGMwHHISDoDYA6DkL+Suh1Jpd3WEGleo4ch1o+OwCwowcwg0l+89

3i9jMMyPNqNqHEuPOxV49w7jOac8pSkVV282iEN1wbqDH/+xYwc/8aNbsP0thf87rhQrQpfLDR6K

AKGX9Cr9BlbtqHNjs0IZw/iMXcOKi+g8uXP1KsBxNtVOIzd7K3AjwBEAz+g69HKL35uxBYDG3OgP

uCV/Jh3Y4h3PFIUoanYhYnMrgEOBfSqjJMcwoRcF1/xAdHZ6DSLwbe0CVyRll0CYIfkiifBjgEyw

cHoX8gKzjwHG3QM9Fkcnoyu/Pz7AikCxN+fEOl2O66nXiZnblly77+N//yTc5vhSCXbLr1r4NW8W

B9EDMPQVTwzfajKqvUYLQkUhJADuBYEeDMu+pFiOW6akPEUMAFoJZHX5PtHcpzJKvGK8fO9clFsn

rUwNQToL64ltj33j3vPl8w2Ld7Q4DkD21Q7s2N4YIVLWZ1L5ZT4AhmOHCUQKkBA9OsIIjwx6sWMe

w3GY8dgFBZAJ7VpxGRggK4KzlRrnAuSJlZiDCZbH44iOBktwm+MLf+PKuVGhHy0zChqUklVMqrWp

f9s8XGVf0CnNwO64cKDUw+NkRrahJY0LCKZggUC+n0NsAqjNq8AYJ2F+tMsbOnMVPEEKVaAhaC8v

h9u1LDxz6eDGiPKD7nXCxBlWePdIAs3kVkDfK8n9EwQjUEb0iAVfhZykbk4ZK4OMwbCAzoLrYAFm

PGaqAOvnjsyXYHQIsZWJQR1rZYhtFcdxbEneOjagthx3uOQT0pU+fflv/r1HbuecKtNTBPm0u3aE Hr

+NgWcHqWnpW9UQXAZrj9Eq8wIVUhKMCh9a/Y4pk9Optx6JJs5VKj0BvSeQMU7g1CEYZJDzJCtZ

9hKHdcR0qZzj8ugmR//JT756tQD+JT3n8JEJMwr6Mcs+qdZq0UMuA8DZpMgCEWm+2K+BIpsFDsnK mw/AjMdMg

+DBoK7G3k00nmX2x1qPt/qfOa/RH/cSYxCyuOCFcfp+d/V2z0mZGBH6INAY8bkHuSpc

wf0rg9JaiCBhhElZXThAYG8gZGGFWo24+Baj1gJVSSdBC06zQq/Ss74cK58eFCL06UFNkJU3Vgt6

XH53Ee5pzuHHf/uVyx986C+q1p2VhJ/AGYJ4rb1KT0ltgQTNElOJ0Ctk4hhAvsKagFHw5T4DTQo/

a3OBrzUOOL9ycA1mNGbqARIeuM+FOvD9zPOTLlrXC3q9gM169WuJA2TvyCCZ3amEKA6GDkj+sDRx

ZXiYPFg04efJ3gS2vU4GkbjAYIGaRZ24DgEK6XcU11eGZ4q76DBNisy0uI0DYb6S3BMcOV4slW6f

wePl1bF3PvZXjk0r/Dre9+h/uVSOfbK8XMuoaN0UX7if5PGR0r+o3L2eN2EMxIHtvSgEeIwlmD+e

g3sBBJtQc5NyjjsdM1UA2sTDlpjSSeayInsrdwgJLrAyBguAG88vE1VCxti9Qh7PDeF2B8KhrA8F

Au4XmJJ7HiIkfDiQJPMg9hMzouJFpIf+uOxNqi552zq/FjEo1zRBcIZIX1KIAaS3p8y8MuXFycB7

3Ew2nXlHEfx3XHrtKtzheN+jr1opRz1WPNIKWYmzBOgkwTrFcgcCj6nk9hsLhF7oZhAUwUo40OOG

HIySz21o7FMxNN39MMMxWwXIc/MV32sbEV+xxeblgk9o13Lmtmav9v6hPvc/R9rIVia6HPzC9/3O

wrTnUwvcys1bUKwa63JMcMCoQ7XqLmiCWx1nh/aHIMkv7qETLCT0gkIAh1vTjaT7ih3bUPMIgf4k

7v0TBISPu1FudBH818xkja73Pfrqtfc9+g0ny7V+dznAitwX0b96usiwBzwdbPBFiAAIjI4wWqwM

vbhABvsHj7sgBMLiMmCGY7YskE9SkS7Nc2gl0IXPR6vinAsFbxovzFluQOp4bAILd27mz7rR4Oy0

p0PjdEqZBp3BtCXQChlgnXBuwodWRmyBrAqi1+pIrSP2anFs/9ncf5p2TjAYzYqulN4SnT3UWM6T

IC6NimsdpaPTWv3aqe3c8elKDN7/6CtXf7Yowpi6w+Vo7yn3apXxPphXJPUGZrldrg3+QE+w1Xlg

DOTVW6AbG3IlGA9hhmOmQXCuM7FIujtQJ1a8VoLOIVeDipXn+b9cQzmW+blc4iAT0zu0+CFAIP7u 3ANf

+tu/f/YVf/DfL9/qXD73nU+cKEHbKeWQpXRYBRWVku4FYiH1L1YNCaXkN3oOCEkid9eaSSOH

JGoVpYJzStulT5uFyoJ1PU9UbyKv1YJeK8J/bLLSM45zJQOcDrz0VIFQC7Vhb93x6J5N+Edv+MJG

2ffVcs4rP7VDw6qlR1sAWr3L+aWqPC8ZL5RrvL+Y0uPlJO4Xwqeeb4uYrRS6eUoGUEzqqzKwh3BL L08GxYJw

+YVeb/lwtjZ7pgpQorw1pDkraZDZX9BbnEI/kypO7cgglZ5b2hT6ohXqPVpPn6UvfNcf

zQ/S3IWDq39rLZ7DkwtX5u99Op0qD3hJrDDGaXyBVYDWAhDBAl8VdGhYU4Xey4ktAISIuft9gnye

LgUsb6UBO9/ElJQXb8eLzaf6tCx7AolvNvJg7sG3/c6r1rbbZRP8wUurMTg9plbqAL2yb6zv1d6l

tPDBhz6/NG3ntqVLrUT5kvws/+xbvlgTaacMtDAtK5xyYHaEJeXiuMQehNDyw0IsgDJIoiBN/scV

4s1wzFQBSpa3nLg3slLh9knr/SWHYlzAAXLXcH/xCtoWneuBam7AOj5oO8R0eryZFr/wHf9mtQjL

E0VCN6AbHMpP58VcH+gWyy80IoALPYTkTbM40ttSmBulFc1FJ9BaFYjxg7FJ0C+BMPct9fvTQKAR

SF2Y9e0HX72FlGnRIFyYLcLld/7Oq7aliM8VmFOY1cvlu4dI6vcp0KekgSuQeEQcltcrP/+mLy6k

3C2/79HpKcfBoFvaHOVTPewuIYDGSGzVW8sXEqxvm1kCjCnPdobmFeQ+FH2ZqQLM1J9gvncNvX+/

Z3THKvA6qd0b1Wp/H6sBqkrgff21y0P7jvQDBZ9c381zkRSeLZbzXHlwpwvzMu9CbUGYxU5Z6U+l

7gJFx4IqXkBpN0xk2WCydbbAKFII+YT6vaQ0qM8ZyLcVBItwa9YXQolDivRqu+P19cY7brIo9S//

4OePYG07STA0T9UYqxTjhugZBYC0/xdHOHqydm774JQ1OHXSSg3ChVFrJ2hsDt8z8pI5sjwFqTKQ

oh3L2+j1ykSd8nzyeA1mOGaqAMV6rZFgd/SWhbJKSyfsTw1uU396owS6lgsYT+J+CYS1jaEcQ3vQ Z

+P1PXXPgifwxcoE3Do7hgeYLDpTiOHQRgRRXtvkkEhBSoAqpc/iwRG8RKJRJrTzXay5hCKgVo6t

Vt/OXRAxK6YxTxOjWv4R0OWywRDRBN7mL4BAPQcmGGCWphOagi3euDF+culNX3xs6Y1fvGUtzlJV

lFprBer5+Ebk3O5lhDPqdVwJWtWzwsvgQTSokPfuBbjtZOitxkwV4ODSo2tFQNe55fgcUkxkjefQ

u6/Nga7P5Vnfnlfw3p9ZG2Zxm8PqBVpcULOlEnAKVgat7e/z6IbRyR4KuIWmkLSyv0WgDYMDgAW6

qNgUvU5G2A+NGyRHgLEuJ8KuW42cPFdh3wu0oPb9zCY8MD9Z1/MrRfir5S+f3ZeF1lWoRhpIA2mw

D573AC3E8zkMilGgBLkIj33gzV9af39Rhg+8Zf3UT79p3eYW1A4Oz47pstb/2MkwSyp3j4PjHkQy

OGbGSCGRQaPA2K2fn/HUyNnGAHXkwRPFYi+gCbeVO2OcC4y2+rquwBjnAg8m5w8T2EotwrsHIZWE

igg5Or4lcmgC7vIDxSl2SSFK4N6DFcrGurC14gDVpvKJV4CA/cVjkATSQsPKYho7DttH9FACiUAT cCABbCsvxsf

+yQ98rsC/7olM4/kb47o+MR6iQDtqdruHyUG9QdJXLrnBW/o+2qt5THg8Uz5eE9s/

85b/v177GozokG3Bp6XHFqbN8gNoeN8+8/dJQ2SJyYTz0gzaEzDjMXMFoBtzn0LoHpA6/y28vxS2

tf7+YJWi3rcn9viU6Y/e9aHuDzqIFoSFIFgVkRERHHR36lMWQzkv+IQMtX5WeWneQ7bzikXm4FEV rM6dZ

+gCvdqhoJxq/XYcPCUy2XWAnkPyIjvF65ZoK0F/eb1CLTWWWFlS9D4NhslVqiB6RWZP+OVa

RTb1Ow5T2OO1q8uyRXljCCHOESHm+0ygEFBwvtcG2bEJJpRNggXzfDw3ICF9CmY8Zl4OnfM9l6xV

uc/pJWWFeiXPwv/jeE7W7LVtNVDmCfIaWOt6vFqCAB4k8gon0cJDmKQCvYAXLLBEXndXLYxki2Pp

gwquzsMl6WQglZ8iZOxNVDjBJqyzl8om/FNMigcQj9F3/3ExOlMykEWrAewzDXS9VEO8BZt6scgy

2URksCf87fiECpPEILTSa7CgWRJgYiwc1yN59pswGhdXBlDoBqYIBiuhd43ukcQj5HwJZjxmrgAH

z59fK8K6au0MtcBNyhqsTbmty6tl0m2OQINDaM1sBz5dMienLCmwL9RLErVbKtnYYPEjbUlBOMyi

WxDrc4PZfXuHBFU0MqEMpcch/hAWp31uVo12lHwbMs1SFNoWpkMUerZdH3kcUPedQmxDPaWRY6PH

EUbV2uwyK+ALZ+keVYAUeRwi7wI6zFHwg7ovoTQFcnGAQT2YGOYa8PcxQlkIz4OfOV398AyrQO1u

wy6Mkq19fKLcQZkeKZXQxScG/vc4LGOUu/50yppFVmuOZulD4AkBnmhsUIfRluSUJU5UUoJ5iezB

IoknQbF0rfTAYodoxVwJiK2l7KsJr0Gitv1omhlhqa+cINeWJVYBZpqMfQLoBcnEjJhCuVC3IxZf

8yKAft0kbiFaXtYT96DK0aN4UqZg5fbE+1hhYYCkldlpfxMFKw98Ich6hRgVV2uEAMAIiLag5czn

A/Pp7cZ4+sD5IsAbENgby/C2qs4OIUxysUkvOawIydWkWPuEujXCaLHBLER7OCmU4IJx8NH6gVli

qSwE6E1s8dhBEzRiaQ2bYmB8QDPEYE2q7BjaGmVSKadbIMOv1WFeQwCsBFoD1D5XJXPFFCjiATBr

EViM41BH8T0qbg9QyARXYJJ6UOFNVUBBBRnECKALcLvH5EqjFr59X4S/xTiCjbQsQjQWgj+6VoRm

FXZh7IoCHFxZ2ijCe0ExPahFDz19UPp1alkEL1PaQa+VSrH8Y+jRmGbdGrWn/D4IddhiNk2YYnS3

zQqHGAGtGaxY51aAlpi61HgiJw+CZYFp8tgCAnbmySk59O3RwC8sc9pKnXca2heoR+ciyEJ04JPz

gSB4CvCAWGnYCE3QmRgTNiFaFKQpS8OwkkzANftNEaqkCaPUozK5PgnB8L6SCgChPEQD6qAQwGUZ

GParnz6yG/CnjpmzQOunl+bhK/ecwtydIuvyJpjei950bS6KM7+8X6goBjq2D7U4xl/XER+ECyvE 4Mp/O

+7EyMtze0P2+hIMU3yooDUzYMLvsQL4b4VRUfid/bEZjTuM5PU+qGwK/yPn3841p54SMmbm 2BWcSrQyBy/

+Q7ki4I4YGhjL9igOw5g1UCYtWH4Cwh5jI/cdhC4CNOhIsk9VMPlcLBXI+WK09qCw rHnO8tG1lPMK7NKYqQKsv

+PhRfjK4Gyx3MMsbdAxwB/F9hXypLrsqK/XRUQDXa6ovS+YsN2nLDcx

ZkZN2OUBqnufWMoUQktBFs6UepZbsLzhUXBWiEjZB7OoXBk6oWAEkf4EXeFdhUstnbv/Ww3zdk35

E/jEffEsdf/JDIArZr32xNhO5CxQjca3q8BPFNixOGfZowbZIOfCJgONklR2NAouFxei0LUkRIXc

H7C4gyGTnxdGKKT7ViWqG6Q8PrZb1r+OmSjA+ruWhkUqPkrjdAyyBro6GaZf66/LFXEcMBdYIml2

SwnBLGV46MECR3gA4Akpoy9Bi99M+Pmhgloso+0apjbhjQwQRUsULXzwOn5eoOcccLhBAt6nkPM7

DW6MJUFszAg7e9OOU6+jQjYCFxqx4ir8ck4IMfDld0Dtiw37ioMX5+wdTmpHaJhQfr0fDfQnRMCw

dwkLNH6QT8wdavTB95tdRHs7pfd8+KMvX4NdHM9ZAYrwLxRy43eLgM/3y5+lb39Y0G4yGJb2KIz5

RwyBBOtq7Xwo2hKBkwdq0ACgV/7AE7ehZ+WzhGoCTNk6JvEeDm+02rO1+O7V+ci+MvaURwNNpWRR

uX4SMGXwSPYzhQOIyu3CD369nmtI0mBKUlHC/ji2jtZWPCj5+YMpbmVtNOmkEBDCNgqRFIJGOlSh I

+hdR/BShlb2gB4QN9o0b6M0CvbUE/D36eFf+8jLti3yW3zHU0fKl07h5uaZledYGvGcguAvv+tn

z5Yrulwgy8HG34dlSf33ICxcPdEWcWywCNu6Xr1MqiSldA1a9LVo+0qhFlrDOaU8VWj5lio9OZZg

eXKxaLDy6W0EH5QtUkuosAEhBNMi8AJBLEYAU4TpGmNlg1fcBSLRBMMDoRZIviPBY0/4USA3MPYH

vjcG8S22ipjfhbdVvjqd6RPc9UQ1ABdrbpSpPgu5rVb7w9dDum/ZidxxQ4zqspY/8tH7lm52j4oA

nCo7WhzPzV1ZfNf1ITyHcccKUIW/TkyRwrWeQGNcxIJCDdBoEAre5ixArsFyJqE4oZfgMo7birQY

x5L34WEl4ZZ8XDdvFKV5BVDB9WAZIsMSA0qLK3pwJ2wv70n3ZvJsaDOyIpxu+QUatP1NSYPqMVDZ

rWgt2TnIvlEFTaGNn6+HvoRhHyTTQ8mdgOJzdDjHYDO55xXILnsO+QK7RxTgJYY9YvxyM0cUShxM KUBJUFz

+1Y9+/dLN7s6PvP2rZ8uZnZDrGY7H+fJzUYI7UoD1KvxUhD+WNHtXZ120ArcsccSZ3x71 Wd8T2g61dketuFpS5f

+zTMDOQsVpcCkZUkk88feUB3dIoVZQ/uaWI2Ddk0PSRTG/17BgmAIJEC2y

KaR6DbQqTfDzl/2lnUsheI0wDB7IIZcem9cLMyqS8wJCR4IJvV2P0J/m3TB4R1duBJvjkAXWZUeN qIGseV29l

+CKxe9kY4iMTuXnQBAUWa8LDVLVgiq6pfC/5Z1faXM/yF1ahbzDG+PxlTcVWAR3MG5b Adb/hw

+cwir8XNZMeVzA9GiOQpDrLU1s4oolxCBWfbaGt4QwUVLANTZWz2LwpiVMLOuJtoCzwQIV 5H4Zslt6f

+gUsPVEVWegPy3zCwEWYaLIzMSHaiUMvoiGLrTdhDTvnAbgRyK5jEnIZ6URoA263EuA eIY

+PKSI58FKFNrfuFH2+DAl+v6yfc2ybnjpssUHqMLtwi/FgOCZcJu9pc9CWDeLK/SJmQdhJQqk RcHx9NZfvQXsqVY

+5fRR0rDCISu0OQiIjz10B57gtoLgyvaUqzsf6vi9Z7/18u96lZ86sb2WR2ed

HzAatA4RGRD7hW0ckSqeZf4bQFuDt7LdKvjtzid132jWSwIpVgQChhNmkdEtO5CWNGel9/Q91OBa

LV6YHtmDUqYcJmAitE6RYnzgNVFGU8QAANCz0n0WBnU2FwD0gtYsgt+DQj2B4ySYnFfC/ODyb7xy

VQ576b0PrZ/v5uBseb2o1lWECzS6CsE5BYHmwFgUgcjPVIfRmyz06O6ivfdkl/N334rqrMJP4/Hl

su1BzS1EVyrXV2Xz8vHF9aOXbiMwntoDtARXOQBaR+cOrc25tzvhjm6a7VWINNY5wlr20NUWecEi

q0AlteLBchseR2085UGgP2iz0BIDAPXrfMito5UPKNNjQo+SMaY++wKeZFNoQT06VryVzEKDWKrg

GLoq7HTVoMKntH0jhphAPBQAGGIGPUZS4G6ew7bqCQvftqtLLvxtfOjRg2s//7GDJ/MmHi4brbAx

AaVXLUCWW4f6UWSFXMGgB5WclCDMVhPUdrH8dXn87TsJf94cXy7ycogVCJw+BQmsySjU4T0H7jkL

tzGmh0BPQ2F8BkOv11FIM+j39AkNrVTooZflHTAf3qBEcPHAd9TYFITA+Ys1EpiiN9lnYEkSBtDu

hXVMjq8xCdWGvj0Ip05eLGfVkajFaXYO/Dpx5aeQJw2/ZkEBHgSTwzAAwe3TDQ2s9Xsew3gBYNsO

IrGOAgtbgMvnhdEKu2DealRF+NDHX3ESOzxclGml7PaaK3JQaAgwSJ5JL1BGUG8BECrpWmEQwOOJ

xv9VhTy3muGlwl/2dQgkviHQOi5P1tnkmvZs6PQPvPtrU7dRn0oBONHVnaLK4oy0tj8yP9K+hFdp bIvVoUx9rPEBr

+ErvD90zUJymQBAyLqSroFl1sM8gVJsWleCZHNnIXiJdov6tTD9rC3IckIYPpPK

z8AuCaQiUSjYEojyXaexQCVLWCWWQm1ia+cm5zCKuODmj0TmA4NCKrHwiD02ZwvtSMET1vugEJHP

1QP65q2OLO0w0f1DxSr/w+IR/tGvHzxctn9reetTdu1GFQueBPe2HPj2BLMGqvW+1sny50sG9PA/

vfjyHbO7levPDHsOQXieogQA8Vz651S4Bjx3/DRN1fBruhhg1F300mbr+0na5ZnbnnSNy+/N87UZ

YB1/hzqMQRobCLPoHnyKsVQPAWDWl62OQA3zHIr9BbeLZRLhjhaUQoeGZDEGW33rxdPYjwCJ2uEZ

2xPmiaBahQrsWDAhpBbDTCX+LVdQFw8JpQ+qiVssun6nXbI6BRTBVAXiu2tUpOD0zU26/L63fPHM

B3/jG3acZPKLHzu4Un7VHzi9uL5QbspC2c8DZW/Dsquh5SCSBtDNAKwXqX+inNoT5UouffjiwVWY

cvzIO586Ua7pXDnPg22hP7IaPqkl0joif/4S5LdSlvIzHOSn6yo4Szsda8dHsv72nysX3F3WEobI 7jC2t5VcEMY

+08szvwNdvJosoNMAslUHoOFCstlNoNYexfopPBILj7qvuhIil+0a9x+DRoQeXm8M

jbAopiAAOqvKa3kgehAwlgnAIZlaPgQM75EWmMU5xX4eafkdv/vq5Vvd71/4ob9Ywkb1QX9Cvgg5

v9bgVwJ1g19ghqVJjEA1AgkYRPij0SiO4pHcpeUPPYd6m22WLtq4k8nri8Vq01Nfq9d+2gJnkvNH N2ha

+qFVAxLYc3KyPsquPb+NMb7k8KXzeMvz2NkD5LlTdSK6Cj/aJPYQ2OawQnv70fV6u9bUqkop

BeFXy06m0ahRkQgVlzWqUDV8FNgVrbhsgtbeM8vO1p40UcQLS6iCNeUgh0r1GJLhBGM0QhCXJ1kU

tzqmPBYsQqwL4sy/ZrRJHuQ03aEbC5o1v6AlBow0jOVph9e2gyASomlguX9JGBlUvh16ZR+ghqhk

VGkEJ37qh9cfwQ7uSBFm0a9/8e1fXaCvfu1iOfMh2fVJUaDJBj937iinNaUKM5EBvcds8ynt7AVu

GQOsL35wWMzjceq3OURLemWZysgFb2StTazfz1wzQ47hDRN7hhO1cZSzJmKtQ2mCJGfQg0qxACgB

EfjUORYAW7WFSBvUMpxoPfpxy2QYYZ8wWnopx5Bafj7nYJXRzj2h7z8E8BoHyDngtCxQyGOglGY3

pTNmq52DB8AWuHMsj+YBZG4ykN43whhMG57mmONEyVM8+ZM/vH7xJyrMeZ5GDXQX3/bUxXIel0X4

I7wLTB9AmLPgxrGqBxNs4glMKarnP3V86daxwA4eYHCcwkovXrM/sbqLKUMXYE+bx+vJGrFcBmtc

4GtC5lK5mGuCyxfKZwtqZbNbQIERpBizGbUQ+AZrrPw/6EQWLW9w+hPAlRKCwEvMACDHa/LEwg0R

kqApKSr8iFbbFDTSuTuKfz2Ryig52yf3AdwCilzUNxII4yPuSwQDFS75+wonHEvr8RwWaS3RYjmB

Ez9xYv2JohAXim1bPb8L5cg1yC03+xRsjhaV4QOPcSLPT2RxG1iWWZwdC73GOuhwCXl9tHl46pma

IV692XncWgHy4IQLe+jaHFqYWKEb5wG82pM6iBleCWZM+Mub60VAHj78J/dPVvwtf3bhs0Majc6V

7Y6D4XrZD5lwS0wQcDkJBheMroFhnsS/7QaGRJvc2wzWekSCRknKBcUggV1ACnkoQiqQB8nnyG9o

0m3KGZFy6ykIO4RkFMjJynZtK2J2S8go6YinVamaOCJVZA8cValJA3lRGP7ekXJdH6039dSJjUvl AJdSGj/

+XJThdMH4X/3q106U8zteLnDB2TawayP7SyAnMgZmcgJYv5N+RgjsfQvpJZcviKFdT4NE

6QTciQKsL56bp83uCGdxD5D26I/dmvsruEhs0Lq2ddiz9oCycLMIf3k5Ho+/+/C/Prptm7tvXf3W tfLrwX/73/7bc

+X3Kc/uiuuP1k0zwgRhMozjeFccscygDx8sewzQ6wAXaFPCWNHZc8/uBcAqIcVG

ewsTAGemeEEL2GFotQSFLfV4fPpSXkDmFc0vaEyljFX7BpoHcQeEQdDZW/SgBslxQF0vwPHy1uvG

5a0fW9y4Vu751WKHHseUrubRaOMAzK+dX7l1sPmuAnW+8tWnngT0E1FYo681lgILdMX9iodH6Y3E

BjUIOn/ODBAwE0T8SOrV15zAyZud100VYPzsYKEjXu291vvI8kVs8Su337h+7vrc+nSOtd9n524K

QjaWjRMHh0QP30z449jsNpc7uqf2nR865TmRQQZ3/WI+lavXrKxsr0G4MUMhYOX3skMd+T6INUcp

F1aBh/4EdLGxVp8iGIVjE9Dgf8oYoII2Nu9qtdVzATiRALBFwUUBE9jWug9+FnoILiURZg0m2BUD

mg47+TcZ9TgsVzYsp3mcV7RP8Ax9ld518suoFvuffvTlWxT9xo3NBUhoZ2PIYPKaJIugC/JRYPhA

vXDD/XJJHNKxJ2Dh18/YSyLNH//Z6wuX/sFLVmGbcVMFKN8+IssbgXZu0+4NSANd5ZGFXyFRzQMo

7GABIQ12g+vdKLBnCaYYR1ePblxZePIowDPzvAd/jJv61PnJh8/urak2eEa2eybu8N7+/iMi2VRJ upfss9S

+35dZLDt5FrYO9eH12Jv19T1bj/PMM0/vSA0mXUmmyaIGtqyY5BjYk00Gc4wpAYHzLtgG oZJV4bjwK

+ThI5lSEVtUeSv4RfV4wVtE70m0vWFDfF0zRHqC+l1Qg2EEV4zB7MggScBKcYLYIdAY

C10AOBYwlKAKXtcVW93utG6uAOO5+72rQ6NAyRvVaqcHaWRV5/HKVEaBPT2rp7Ai81N8HG5jHF09

XIVmpg1R9/KQoF+TaqCqRSE4z80nhdgiKIFiiPC3mgikieQdv28BJyqJYLBEqkDNeEFSXwdKX8s5

k55oeWvtJtc1lMPI+cgxVZj7MU7/t8BNTDLhKEnZfBJA2KAQiicApUUJOR9Qv3fTUumbB8F5buhW XoNcq

+mxSe61urOVRNsEblQosAVXNteGNNP21i/CYQyWQ4Q6DEPWubJiLgkicwL6HRSr78KMKkgQ

dgYGh0TYZXvxCKpIYfeECkmAO+FxvR5o9r5KY74G24/72UuooOtl9eXEkl8BfvGaCyDUJn+HBR1E

+KHRoGReoJV92fmNgW6qADfnJcbdvHRsCw2sIscv/f8rPEIGW73ZTODLkJLEWuMpcfBf5pE1kIMo

FB5zsGCzP2Ca2OIsMgGCvgWVbCq3PEe3wO2fpJBf64wiU6bkouUROKYiLRJ14QUhDSrEnbymxcX1

eZKFu/m4W4kK6MMeHiLcLths4e29qhj17w7Q+X8wz9A6fKQWptw0F3ArD3BIWR4M3d10hpe1LqSk

9fdWV2NuXIyPlAvrxd7RzJ2/NCMDqAeowzF4z7rre5zxEuEwQVZvIZDGA06GLq4gUVk0igrUqSQV

zfu0AjtRChNYdLq1x12Fce+98zAayfGMkbJtHVIlO35jfbDVXomFF9ZHLX1ShWxBvzA+4hESL8GE

0mCkvHf7ChCnNfKKL7bQnU54QWnjbTOwAH0xOadAuUzBSlYBFp5ceHL+MGP7HceVkhMot244GvRP

dxS2GYWP+u/fZPuJqx7JrRjVV4MBvzHY5hi9493iXLY5L34xWjvz6KvX4BYjJzBwLhaagQvLoeP0 /uf

+N1hRoAgtUh8SGeXDeJp6ATTKLizklS4XW5gnFr+YnyCJXXCrh3+mUAmDOfEQrEScCvKyBmBR

tWO0RFZyPp8U5jjNKUoAPBdC/tZ7lBpE49rAygTBTcYtPEDXK3yTBaxJZndRm9CCBHHhaArlAC3p

qG5PGSGGQ/c9O/7a2bL5GdhhFAZofhOfvUw1gBrzs5WLkgfEV1fv5ngsVhOtSA1aV1AAMBybhE/O

5qnM/eY2Zb2jPK5nWZiYbIxHO2+hHjS3VQ6Y7ThNmNgqmrWW+QI8U4tlrgh3ehimqFBUChJcEdqu PO8Aka2Rey

+CwWIv+zHB7LE+gEoR++fttZNBmmUG9QQAMY9g9KScjzJN7ZgHJ69npSTPfuQdT4kU

93bGGkjIgip7ArH8KPdUi9x6MYTlAESZJBGm0Ij3I8p/iwTkzT+aaF8SSx+y7FnnxFrBF4Bx8toH UDskeKVlu

+BTn/2uz56AHcZmunGu7OMQV/3VEoEss67AjldPQOfJygPRCk1XkiYc3qJQrGiTHUKw

WVg2wQZ1GiRQnHmmtUB2vb0fz0jr9UqQJotc43RdIbJY3AAoEDRrzSjTcLLWAPEvI9n52vqWmGOA

AH9UeAHj/GiQZ4oT26JEHZZoFGVCS6bxtvU07tvusqiuH4cRfmnMEMoc9LwMx4MKtUAfEn4aDbEx

/JZnmtCCYrH87D3SNl5Jxq1igLXiBYaiCJL97dAyoq0qj1zw2QoC6MOCmAmVzx0elTuRL/7Zd312

+LJ074VJOPSvFz67UL57tviQB1yxhGGy/QL0s76S/jcrZwGX03YgE9bl/Ntlyr2Pk2iicplSAfQ8

jxeXgV1nLMXudW2Q2iR/9LcYAoMpBpgQIYpgaLHnmonOKoSo3H+49xQFWQQEyATOFMctOWGPnWHR

BtmzKGGAXuL52GBsG+OVba6Wbw89vrD7inyKtaiQMUQT2BQoUuX6JRPcvp8Sq16rAjXF5FJoYGUQ

/F/P/xrcZNxcAUaDjZYAG2l3N04NiSA7TpRnbIIHfUsY4BF/FlIqBXYsfZmun77ywL9bLe9eLZ5l

vt5AKYZzQespUpNpzczqSpDR8utDwsgyaKGaeQE9F3RLxIqgWDYkhaIwoHtyeY0keFqVvvU4svuD

LmxTeAAumcaeqmgcbFZPUHOAQSTY3hVXhJYMY1tKwO6XQyI7fz4QorYsJwukeTt9xqpgdlz+u+1g

uO2FUX6iePHXAVj5he9HOXw0G8fHY1iDTeySnH8KcQ3HBfxMOo2SZK/otCjcYm3hmz6Rwvas+XKl

Nb85AGslCFJdye5rAuIo9EFVXyLDb1KKizbJvP5zXwE2x8vfS8WCnnbhdwUSgfV6oFb8pR0l9Bhg

7lRjA4LQrwd69T5gUEqFjbSnfyh8s2uBUKLNffr13HTCic1DmPBQqpAKNaYaqIKh1tUTUc7VqwAS

REEGU9ykmyptiQ6b5KYCcAEhOb7n90gMiqh4UoEP27BwceCaHL6U/c/fpFHVqnk0xL5hRIdCOi3W Al

+GQXYc4JRTSIShevuQBwAI+6ufr8FNxs1NEg2uoazW0pYkBcFvCaMbBIsFyLsuhIZLNOaLQ4VA

SoNpCTMLRSKDGdB78IbRw+rtXLNDEFd05CBbrRbYjVABxniuKpwCBdprPk8wCk72C3FtYVEyU46M

bsH43gS6V1wyyx8bjgQ7YyAKP6ABL6kRVuEmLddxa+7nCJLGMuUlNI/R82TmadxDOAa3Ugp0D7DN

d/Xaokcaj8ZbJqX/xke+frV8+uXolTWGkQAerIgt1vYLrifn/5l40DhAvtc8RGrN04jZI2IP0vog

3Hx1yZsqQJfvvawrteTY6wZMuzBObpHuCR4jgN0obmNo27OguaAwjOlNlwSdWOIwgjniZGt2ZRNg

C37V4rbvjG1/bT9i8UUoQkdock8FDu8QdOK9C6T06JQ8keN+v9YAv8BnrDFgIg+sdxwmiCDW2AJO

y634s1BLA6wYYNcS9+PGhTtgyD5AFIW9iHkS9y6gtpeVBD12avvyXWH4Tv37ddtdV26LXCf3NnJV

oAqKDHFMCROhB70s4GLd25wJM8YodwA1hgAQCpWfBI5uWn1wcw/w9OBxysmsW8ZeMgticOsCn6yr

AticXtVyfQAhNkgJIia3bRD6bA9qVzgxqUlbj6CzTeIF+OaIcAJFYXSWCKIiW6k19N5PojB2LST7

l16intfQ4NdkkVrnZiCf1M6/YZpR7gnGvkI2W9SSUiaQbnnJPEamvpuJFh56lpvM0PD3sMfQ1CPL

EklsdcEE3rLPKqzZiSCNQ47U7O+Wa0McWoDuQs0CqwpojA4EOpOMHm0/zUMgWLzAVl7ZITRlFg9B

Xbp9BTh49WgNHFajdXOMz+490n3KiqiGe4PasPCcYTPB6MIls/eQld9bqYReOLogi/fI+pARwFds

DEINrpwGZcjOJyiM7TtCK4Mu0ZqP5WYyU4Tk+xQ4pFYXdTqkstUKP9pnU3mAnCFO2OeA30CMC6/u

e9LKN2gAvXtB9jeB3VcWWrbvnkHWc3ZLb8GqCFuwt+00NN1Goo/y1vzm3Ny5eF0/9M422X3Y7i3f

V1I44xYfGAYxbifzTFKaS27dRWnM2KJCP4rCz503r376zEvW4CbjljPCyk1YLSe7YC5XmQYRBglm

dCFndbGM9+VuOqyY8CDoE0e4rQcBN7+Vrs98x6PChaQb02f1I5lk4pAk4FxLivGT5DYjFK193UsK DIrRj85+cH

+diJ0x8uxinnu1OnXojC47l9YbRz681dBkllpKIAhBo+YD/BnpMRWGkLE5IpgitqDe

RECNKC9IuQOF/ZFs1coMkLQqVPbD0AhIhIxvvz1r8gB+8Y3v+NpC+ata3weotTXkZ4KW2QWHLNrS

RCAOCq535dDYUa8KpRYI3frXrVPwCPy0HoFb3u5bjKdv1KapAW86zgbG9qKVwkMLC6LtKqyrmmNq

s9TkCRANdOVhaFANfS7crHyAU2qZ1WK1oFRdqzE/HlCJlSaI3gg0yAYk0LbmKN7ClNo8mH5e71zo

PBcUToWWrw2CJ5gmCAbwhlZ2zhAEj98Kii0CqwaHCRWLnTDESgAUYzRRswnhN5aoxQhowTZhH++j

GQyFuHyfzTg0jz0sf1aG72C7b/3kVit70u+DPydSlifKSUved+IdJNCV8/YAXyfJeExRz+KWfY9u

qQCHCwwqD391C05HCB0UXEv1fQBlJpR1QbJ25D04kMhWVMR+nOExgP0guMKgKQ/044pejKFeQYJj

EVTu4wkTNCtKQIsR60JYt9dyBSBNuUzIqIerKRgLlgK10qO0swfI3P/UvuOeh7ZkZz0jDYbTDa6i B8ZClwbP6J

+rl8oCJQHca4lTZI7RPEwvVmDvJ5gsGzMV7isonJWfFtiiMTegc3nFwoscKDwSVoi9 Q3ZkxgLetfXSmCES5crJlLI

+r9VbwR+AKTIzZWfLLETOm0OvQzH2+3nKjZPt0eAOkUCdIKzYL5OI

CmMPgmTB5ybEaFjeSq8DX9/LDKMKJsMlyVJrIA0u9LKgdaTyWOhRsiwU9+3xCCtzgD69OKSdes97

TmH8+ZGQWTAAX6Ul1PLYj8EduSA0QXVj1aSY0KxyOJMQBCs0sl5GZlWhxyihZqMVGYiMsNvhLLue

J1HQd4U5rrAkhpPUM/DEfp/qiIEatWmOWSAThMBYYgMC9wC8Txovw053e6cNXvunR1fLQVb5wpS7

J0tCseMK8MYCQfMIXDJRU9cJsVeP075q2WVf0BpVUD1o9R+K3sLpTL7XFKyywZ/A8JC2VjfXLfrE

+xPFAg2EhZ4VSGasFIBAOnCxRjcOYFjYeEKyoHLHkf2aZL/gQuXCSSDmUAJZcOts3oF61twV3xJf

DO7FYFCsyddj6X1T0kKuCpRyNWUioYzAPA0CuDHie0Be2VmZsob55bgi8O2+mdCjxQrkXAMoexTi

BFEF9pTcexlW/+DM9vOA45iiOqtgbaJlETbyGymCDWKhky1sx5aS0C0MBisCHthy0yky1ycUZi9D

K6ujByhkCTR55higDlsiZ3Ca51CLxVWdKMk5UHbJ4pTAXsm6AejnoWv+8jbmiUhhiEMiAjBhRy3d

FS8xZXt0tbDyOpRVkOJ9CvdaIahaXn42QAotbVYYyTbmKcV76oOBqDDovsLvEUSP0KNSUWBUUCIt

WxDrLkqmnoCPqFZfYr8g5Kx427I/Wv4QSyeU95f4AEfjkzDFmEoBDhcvUB7BBQ+C3UKblRXzoHAF

0INYFnQNNtFahxvkEetLYNtjpFgpsDmmUMLHh2QWiScBttZCzYkHGIunylsCawnc1YKQL57RC3yB

HzCYIRYakcSVg9pUb9zbPAhNC33skZAGgCZ4OoIXMKusP+SCmx0ukZRPUMzFgCgveZrEjJsquXge

CMePF98suOYJ7OrVeMgXMguwQBmSbK2+JpRAln+SQqLA6ojlZ1tqsQK5IlmxBkMhzwM8shP2t7sN U450IC

+Vy7ymkMFgjLFCyQQwCjEEC6HFWYic0Y3CCGgMS3DZKthkOFNvsHgijO695ykkwFYr2ihT

PqcYOKNBqVCnFIL8PsvUIBGYcIOUWSvuCIYU1Ov06l6m6Q1aTcoEdMkm5Aod5UCSGCKhXAHUAzXb

KtCA0batXqmHCZY66wnL8+kZCFKPbu/LGmISC+hz6d1qschS0gBa32O4HT351THkaf+JYij0EVYH

bd3hZGxTbAwGUvIsSgtrXTe3DFOOqRXg8OrRjXKQk/JAOA8ADhuUzVFMp0JpC0yociRvGhtxdcwR ZIM7YF6Fg

+HIDk3U95v71+7OpAwFPzQPLJHAqVZwK695xiB4BP0gMFw3yXfEjBqMIj0nMo+mQpen nA/NqA

+jcWA/i4zZLRMrbiJ6SLPuoogOgTSOYMo6S4RkHkQVSFSJvMwhWndQvG0WGMA9TPuecRQY

KGuDPqqSCtlABd1qfqSsPgS0TZ010mleQM6rzQfGXjKNOnjw0+++efHb5JhaAer4ls9822o5kTMK

XcYa8Dg2tJoXdcc9oQZVHmV/WGm85AGhh08hKEIiYyQALUsKlmWW/eXICE3Sq+SQyz/z/Xo+wGuC

wrmZa2+PQwwoBZigNjcLdtCgUz5rVQ473eNRlRyDPmiCp1bYrXqcEC+KHZQXQGhJ/pvhmuzSvHFU

GnLvQGoo7N5gTxlE0HVb8czKxjAUEaVSihIpFK0Z/ZmC4WjxQAtgVVkYFaS23C0/tzodpVP2D0Je

of1U5Vn6lz964La6jtyWAtTx1z7zbRdyyg+7gARrD5LQEoGxgFUZCZ2k0hP28BBR5qlaERpvw55C

4g6JFbJQq2K1UcuvfT9gQm3xgSkHqBKiCrAKK+n5yrWQ4HtS2UU/JzK8g56H0vjBhMUYM5qmFCKl

HKhH6gk0hMBbFUHe54PLtYK4NnDuX7zh1v2pVwH1fghS3qD7NLtPZuGhF1yL71TvL99NBm2AnNqM

1h8E34MHtMgz21og2wRaIBCopLIv4LwAgM4Z7tolPPwHPzo99LH7DXcw/voffdtSSdcsBxfXw+9j

skQRgQullxnYA9blPlFvvFtoE3gyupMfiAV/JlTZvI8Gha5QvSwlgCsRuoVW62dxi1tuMgim5ynu

PHDlJuSx5fuEYgDBzta/jmz0FkGkToP1VqHWSSKgsQEf0pVcvCVvCxCNDVls5d5P8w3huuz+iRIF

uMF5nQC1hJlJSFsFG4zSDNbahJu46hO92rMX3IK1PwlFc+BKVLdf/oMfu2cJ7mDckQLUcX9RgkqP

mtVBF2bL7ALENXvRvAECbIVJyGtjUeT1PTvL2F4ZoyBoKJDVoQ8GxsgK9lRBwRWrHTvrqpIhLrAY

I7AwLsgN29S5BxhhAfQER18ThqAVpvEA2u6DekKmx0awbLMchHRyeFB8t948bbVdlAW95iF6gStK

lzctW1GyI3gVOaaVi1hsYsJqUxrFalsbE+wrhM3uEqQAJuxIBtEA3MeRskGWOTZDmWjpD370nmW4

w3HHClDH0T/668UTwJki+F8WrAgShAL08KlaEbakBp/EpYcJNHHCjVtoBGsYJQFssMzg/L88DEnU

MTeCfVjUUzxEkiasvtiGU7du2jDSjBq72DkTWBDJTwv0e9CzulMN86RyXIU9kR1TCJSFH5uESqFO

H5hWlZgpKehBtaLOojWvrYyb2V73cLI/slmReh7106Q8P3Cxm0Jg0iyuKgn5NEeSMgdAn+SOZu1J

mB7qNGkp0Ii9R/29UbY7+VyEv90deI7jv1n9r89vAh4dN4rUEzkNtiSxQg4V9MG6IHICn9hD8JOL

ga0KK5cfS5CLsa2HbNvsmjJN6il6+wqC6hNrevGCBcchrgnZaRAlBrBaHT6JYPkJDIKo1XXIMtXQ Lnu

+L8by5IpLFgeAQ0xRdlIWTpQ2ggk5g7qnEA9QEGy7VoWkkS5V4KHeQcpCJCwQwU3Cf4XJLG69

QWlRiNlfey9pHRA6CwRknR1UEcrHT3Rzc0dLwLsCz3FMt0rkDuNvcT//w3+68O9OF6E/VS7gEPAN

F4a6FytgH3siUGRuwKycNjay91mApdEWQEiCNWFjWIJKmfqj58/BAuyYnyAQ5opC4GnHa9uJBUSx

6mQxCoBbR0ZAfH4mlKDwgd/LjmZuOjIIxrJOgtiz7gatWNAleRdgjQoNohscXohQvAmwtDo0Q/8t

zksOxddI4TpUmXn/qHDH4Ip4BCt3RsfsTl2iKBjfK0xCZfCRUCptIcQDYL9xo1zJ8u//2L3nYUbj OXuAOL5z9a

+dHwE8YgmjELCpJQaNEQB8WmNgg3SyfRaXqJ4BSHv02IQbs+ykM7bQ6DfwAF1jDgap

HKcQGiukwpyMlbIamgiXwKCVCQwFOpTfaP8i9C2+ZaRxytsdaM8ANyAksRgNapTuMEmPZ4ZE5zoo

gGlQB8lPFWKyEXpQS44hntVgEABoeUMWh8PBL19pmN2llr9Xrcl/awmErPajwi61/JlPSSfGCCxK

VZHP//7fn53wA8xYAeooCvyAQhY2VC25wfx4gCW9tbtIBTbgf4DWDE6flCfUCEN8gR4UOxTQWEQ5 bAk

+yRXC5hubYsZEnEILZAHih0kAAfdHiAMBNkS2CnoKCTR1EOyDrCQ8B+scaVn3DnSt3PMzY8pH

c251MGu92ECuXSaTWxAsITWSe0TZLzkuBxF6AA92NeAFpMDOgBk4OQ62Zy9Btfb7SYkgWHeDSxqb

xP3Z66YqCzDjMRMIFEe5SUfY0rPFGAuGyMIyGP1IEGCQfdetNqg77/H/GqQa1DH4xFZMBEZoScUl

oBlcFSZwaATggouMAijCAXJhtmtsCsFQSXaPysxowGq4XXARH32KUogsNUfspchwFRJ6XCRW3DB5 fTMf+1Vfv

+vq29/+1bVyAy6b8sr9UyzjNT8YrjNYecEkTgszzMTe8kPA+QFrQehUJ0MhhkTczz8q

lHD5AqVkGzZt8lyYKSJ0KNSM68wbK8/UA/zxf/fvi/DDfdZiUN4nY1ZCgNnjhYMikFOYjJuDYghN

qULcgmJx3lyOESy/GEn5PokHArN0ehw0wRBIQ5aF9dhEra7+ByY4DMvIPAuZMigkQVPCaUbfSxC6

wEusQwxt3PK3969+eGLxuo/UNiR1SquyWHLWwjK5x4sKLqeokI9UxAGsoK2dkRepSWDbE35OYqFa

efFe9bWWLShcah13qqIkA2lNMTo53yQen+ONSnnOL/wKDWGGY6YeoNzseQlWzGJmpZAjdu63JdHg

DBXzq3UOLI66fgTpVgyGqxVi8Osc8LAFu4mD3Pa5uvuk9f/gwaQKsuId/Rw0sCWwPIdYfIgei2UJ

FaPLl8VjwJQq0PaANrvKMThoPMOnzC3AASAGCr0xGo9Opm5wpRolFWo38nyq/t3AdoHDpnZRVpfT

wvONcg8updrJL9G1OUhX6zaP/oJXXx5/7/VhEd5z5VkcNxiTJKJn4bY6fjKPABL4ynlYGbR4P/YE

2OUb1QuswYzGTBVghDREkhbpBJoP6EGaxqFTotCAFgli17bgDRANjCr2z2ZV2vYb5ZOVMXHjo2J0

7i/m43i5mUMis9wKQVDrfEgYpuYV1NfLeYIcV7E2qJIJ7Kq7aBPeBXIZtOAtRYjUUqPLffNAUzjc

BKpYDu1s79juqmN1oX8KNHjX4vpC8QKrcVe1K/PiW79ypmxwsW0NCisFk2GIfvUc2xkLMyP3sPze

yJDOQwePf/Ifv3R1p0u49KGmDA/+3fc/84flSMfA8w4Ga0AJhEBvakzQlkKSbdjyg8UGaa6bhxmO

2cYAzQOABqT2AG2GGAB48MbYzzC74Pn+EqMkrUeMmrTPy6cXEEZLJy9tWWfgzD95/f93ujyws2X7

ecf6TFZntdVgMUKwgsFTGfMJEsHx0OZarNsISnM6nFDFAFOf8B5NdRvBlAfURmY7LT2mQaN2Orkb nIVtFoJb

+ejLVxbf9pWNsl1RApw37yLuAJSmFtiG6A3EsC1iDhdu3PvS85fO422v01YEtxgmPMbs

kKwvJnU85tnQ8L5MagnxQa/lOTBDRPk+mOGYNQt0nxS+EVtrddlcb9PH8xE6oDbDgkiJRirTsD27 zOWTl157ehvhb

+PvffKvnC+He7DeRSLr8WMTcSQobkPS/qoWqOohLBGqJwF05W1ewI6WQ6pJoY9E

c6b0hjt2REHZOqeR2kTF5OgwxQNZL0OGB0rgu7DdPlf+2csv4dzgaPneSksHyJ5JTikLpgIhEcqr 9ZRg

+dmXvvTwb3/465buRPh510XhtFYfCB1Vgk6OIRV+LXazIBkFennRW/t8fIvVXu5kzDYGSPVs

mePVGWJiUWXGlQmaQRwVdAA1j4zP29JLumP0aY5ldyuLl16ztNO5/NgnX7164fV/sUy1zbrHDHwA

cBdscQVGWGMHJjKmHzlMbnOUDVYJ0Q3+LbsW3pd6Oha0nYlQ20LgI4QzqpYb9d4iA3IK7FDZ7Hff

9a7r3/7hD2+dDbXC751cfNf15dq7k+lqmi8/Q2i9+3EtUSq4fvzEox+ufTyf2zi+dH24CXkBq0wI

a5ST2jKIhW0tBlMLr15XbJ7wZhpXQqspn+WYMQQaX8ttkrtjef1xqBGCVwhWFxVCROZFk1j6d62X

z49Mezp5dOACHdg8LVDIAm8I+2+vG95FbWXY3iI7ouEWrnMncMyqGioCKrikH3voegRti2kfnuJ+

4rWuQKtUlQ2S4Fd5dLmjOeP8s5vjy4uL60cL/t/WaosinJef5zSOL9E83NgcluMPx5jn2yLJ9XQG

6dAY6HSx3PdZzKFWXOCwvybOD/Bl8rOPFKlmkEVpigKswQzHTBWgJq6YclPMj5q1VTPo3RfQ6/O1

fogJReXzKcYNrZtDuTEb73jsNavTns+ZSwc3fvENn79avrqgSmWJUWVZUPGoB7SM3eU6+NQsKCbN

Z0BQbBX0GLAHOKQlITjVjLCqhsz7yZcs/mAFI4FA6AoHEEqii0Xv5q4US39s5cPTzYu9nfH6n7i+ MEr4unIGx

+mZZ4YgHd20W3PmRuJaVs55Z63bNyEW79gp1hfatNkHAZQJwDtCk02igXzjyzDDMeNE

2NxV4geoWN8wtb8WV2aQg2+M0Z2eAAIQCGUTV+6A/sotIyrehg09qhcyG49uzFV4yey6Qpn2lgk/

WPEY6T5BroqYuREl4MtgipemmRMsMLJpE68HgJJbiEpp5ysUKajragpJh2BcPMHbv3py5SPPHc7U

8f3veWqxHOvUGOlIvazMgq9+SAQVAdy6sxlBeaYpUY8BUmHnZ6BXCEKPksIg7syPTioP0kwXTZ+t B

+gKayCoQQIz8OSQWHJx41yU5rXlwgq0B5x7CShToPr5bTMAk8m4aE01CCSLCKruJlsXTD6GYG1D

iTUFaONxgCN2JM2utt5HpPZ6h1EtH7dAI40sMgQYFnaSAwMl1wieJIFD5fwu//A7nlrpum75Tr1B

5fS7TbhYzuEBCC0LQeAXaRDriTR0jE+uGGLVHQJhKI1u2+gaAOi5JPF3YQ4wvmww04XWZxpRfO+n

D6+VC1mXGAAzQm+Obg5JMs1IGvZHm5NrsIf0b//OwYvHn7wtFqAccIMCBJM3Ici0/G1ZXJxQEg0K

5PNm3fsKhBZTkP/YgwZ3Hjvf7tod2nCz3Ae5Z+R2nveL5nlcEcFoaOIOGESLm+P8/775HU9dfOPb vnL8+Gma

+v5VuJOezVdY+NFKlnWOL4WAVhao4HaFej8QJdvrcCbrvUhidBTfIzk3pfvEvvCX3xur D+JMPcDMi

+HKxT0+xvjw0eYGM/3oHD9ZYsqmNRpDZI2nhA6QXjvz17t7FuA2xk/+1ivPlH0vE6iT

9XULyANxp279QojZJwJLiGmNT/AKLWZwyRTHL1aQd9Imr7fWkNOcME/8Ac8DAPaxPoAnC9G9JGph

m8NHVpwkXgIXSwD52Euefnr9De/82pU3vPPpx37ondfP3uw0fuDHb5woyviHBbrcx5afet06AKXQ

EUlq9Ym7Mqull1VayP0CW3CfDG84Py580RSpToIxhRMFakqBqzDjMXMFKA/5cQCv53E4EYQNuGOb

Bm6KHXVCilJeNotLkzVs4M7drhd436OvXMo0PlOhSBbRJMf4IPEAhgSWegMQYZRgF+y65KvgZRSq

QwG2gXoJZTR2jgGaB1BUDwGyKSbGuF8w6AhyX2ECPoIooXk5fu8I1SA2wbXtzqHCHqLN8+Q1PmT4

3qY6qkEQbdMIxIvX0L2GPU9yBQKHTfW9LgTKEgnytuTZYciPw4zHzBVgk+CSlz2TBcQ2+R1UkMha

kChECgoifXdI2ZjgQWj49Ny95273vH76t77xfMZ8tOxwzTxQsKCNtWxbCuQAtfJBcbdAjnC+GLPB

YYTrnu52J/Dsst8TVkBL2pH8gMUjpNcEXugHnutoHlQVmD3Y6u98+CUrk0evECk9S5epJbEmMrHW

tY1JAUqesOLfguN9bS7/blj1Xa161n0LXohtTgDdM6gCdTh3CWY8Zq4AD9Y4gHxFQMX2Jkya/taK RsX

+ADE2iLhfbo9WQNaWfLh44Qf//MlzD31uYfL4546v39Q7/MwnXnUVafNY2dkaKDsRzpMVjdD6

lQJ6sNf2YPAtULSSWaXgufrejtSe59zXjW1H4hZe1Msky7EtwGYPVs8xy3mS07juP8RrCaXKtK+e

G27fOTml62drLZVCHG1vwtnaIPBxeqMJvgk8H7NrkEYmuAQBh1AH1KWwLVD0GNCLEdKV1TdO3/Bq 2jHz

+QB1FCZltWj3gkIgFwy1hlbmLH04Q9AICD3YhKHGX+EKC9UQxunyL/7QX6zltgoJbhQLd2QT N+//h2/8woWf

+s1Xntnu3N736KvXlh5aP9ZhfgzCos6egZW4BahXx9OHHPJarDEIBAgCH5SG5Pqm

hECtJTuZIoknkGNoxhqsSpRdF1qLFL4GgUHqMXg6pHiSClPSk5/8la9bnTx2q+LchNM6KZ1BoRiB

uNhJkmCqk0pNtdgNEMp9kHyAQpz+ai+6vUDOFAyJ7r8pjOdqyr37JdiFMfsguIx8IF8oJ71hqzIC

OPaH1mlZcLeURRu7MrHqDGia3595D1rxdsPy2evKNifK30eku8PpD73hC1c++ND6cLvzW3r04Nr7

P/GKoyUOOd/H6f3X7ZXFBVa1GKAaqSU24QfJY2SDK/qzs/HX4ShKa38It+B++e3HjHDM4haJbtAY

FRaq8cPbHTdtwlmNZzwRRahLlYJx/2C1+lk7V/SCVbmHUtWpuQqHSayw1gGilTy3+RwyvZkmhL/A

n3G3CrswdkUBHrx0uAh/ES55EGb12cJHTE9h2qA8bHSc6tliX6URfB/sYcB7kwqLKW76CMH48s2U

oI6lj7+ieAlaJutnKormcKhtZ8ElgSwu53GCKw2wsCj6CLkPUYqpWCAv++Bj6Hn1FAEDHgeBS2CK

iRpGy99EwWDU8+0ObBWmFvginNACNYjHcGpSC/8myhn4giHpZBjoT4ivj6VLvcSZsUcxMWaUpypv

2099xiu7AX8AdkkB6hgM8oVyARsmTGbV9YHpA2SGQtqZiEdwipEFGtF68qOuFuOWWA0lSQpFI8Ty

13CMoys/99AXFm52nkufeOVSARZvLSexTmGfJIS0eRoy/Wj/CvYGVUqDHNKPKOupQYAeU40sIseF

Id7DHCzA1gBXoI4dR5SSJpXH4nYW3scfPb81KdaNixe1vkFcgyQBvFKSZo2Z4sQwaQUsicWFbRw8

u7dI8nQQbCKMlj5XL2KLWwNSCvtlmLQxN+6WYZfGrsQAdVQv8Fvf9x8K80JLJugEsT9n0/Cxgmrv

7CCFaUZNgpdWoAmizg9Q7KkQC0wnQOOIymNf/rk3fWmpwJ5tb+TSJ75hpfxagT0yojCrkDdXlzgo

EoXQuMQSjfZ9ubfcTDdgdBbgT213zDHm0+AC6j9dew7C4Gh2loLVV6JA4wVVFEneJXmCSTyEsGIg

zy4YFTE27kLZO+Rf2i3rD7CLHqCON/zea5fLBa4Zdhc2Is76op41588zBpYFcUvp9NjgUrSIhjPV

YYOwNu2D8v/ZogRnYc8PmUsr6p31clCkBc3yt+Fze4XmtAAYeM0uKzkQq0u4pZSgBb81lnLaU2Ck

3UaHLVK6gGapKXR0IOxVb2r8kcJ3698xO6wK1yXLVUhWuZ7Dtc/84NwS7OLYVQWoo2QSHwTH+Ei+

XoB1HDCsj9DvzoZOnYbJNM4QheAYQnzBkAVDgRt7ixJ8Ly296YuPLS2uz3RSxUxHUpYLZT0ztpkO

CTxfkdUbgscMCskqBGvzsaXliDIrn/zH26ybRQX+aFZX+nmidXbGMJEdwN5TeGPvgXD7QNSHMOie

BcGK4ERppAt0yCDrdwnnNtMx2OWx6wrw5kuvuVow3ns0DlB4qwtWWAkEaQuOkPyBmISKD1/EQpFw

eF8iNp9zQFsyq8fzJv1fRQmGsAeHGAUVLTCooV5N7pvAHKVI2yALO8t7HUT6uVrsOp7c7phUy8Wj

krFASzKLHPenZr3jOcQftfaB+gSvBcJwLbHMOQTUBp14IvzybkIfHbuuAHX8yKXXlFigLbpN2r7E gkYAY4kA

+ivGxEDX5wz3EmvQy4Bimy2FRl1qUIyOk4VbPzzezJf3ohK0UggCICcJNBCmXpAOQviA

YmfxHGI9Qbq3aZFa222iww9VuBMGd3DIR0zAE6niEKiSIdiqLUZnirCCC7gGsvqZ5gYleYbkUMi/

r0myLOxT/V3+/aXP/MDuQh8dCM/j+Ojx/7hShP+EQRQMOQC01SHltVovF9waJCOaV8DoHYT+RKsG Fg

+QRRmsdFgWt1PhKQ9lpdz5x3mbMDp/OYbtX+dt3qtfHNu7HW+TeOMxjK/+2srBW5bzLr71q2fL qS2pxwOwa

+K5uxrXBHIA4n1gSKmrU/YZHBa+NZrDY7VzQ53RRTeuXyxSenyyn6fDHRBrLu/HievM

GIHFDqHcIWP4nq3hi3JO0GeCYik00CN//PrBIjxPY9dYoO3GWy+9ZvFXj/+niukXbf1fZnYCxtdV

YkCywxitHjCTVN9NVn/DQiFYkxQLq62CwJCISdXYgiPuE+XlicAoseDloGD1TFISBso4fZVLzn4S

6NH4J84BIBaeAqwfLn/fup49OfyTfwhsQojAi1COrdliUAalCZ0kqJKIVE+YcVhO/sli+dfo2WeG

urhc21fy3IKxO87nW2DrnZrJkli9xJV4AFUGD9qb0rJCaxYb1VO3TVaK8J+E53E8LxAojnde+qaT

GWxhDZDYAC1htiUIVsUgE1C2hmTpVbX61iWOJU5CSIEQxNy2W0wpMUAxq7ZvsJgkriPKsYSdM0ua

unfZxryKPG/ZP0ZMv9PI8khcwPtZYGZ6ECZoTwhYnT+zWp2meeSNZkGL14Y9xWDGp03G7CWsRH9D

IB3YHrkHyWCYJ8DET2VNpnV6DqTfA5skXxe9LrDnj1/fPa/CX8fzrgB1vPt3v2mp3JQzxWqsW2Cr

0MYrFgNuB4Y4LHQAYMmzCIPstqvCuAIhp/HByxRkHyi5g76XUaHtZVjJ6EfSEA5FISgWvwFZcCre

Qs6PYIo5weU4XybzVAA6GUauLVhTZnrU85FtSyGjCgJNyJJYfaGHyM4E/A5GR1Z6sgbQWr7QY3hi

cBspTasI1RyBKHLrBo11JUfS8ueyjzrB5fRnXj93Gl6A8YIoQB21d8+NTfr2Yv2vgU1DZMrTg2GB H572F4/RAAUFj

+ClzMoCgVtNQPsAnEe3PRqM4l/R2ivE4I/jfABRVt2DuHU+Vg7UrAWvfMpPwA6j fP3J

+p0MEK5BFImwVwqilrRdQZdEGDEIMYCWeQDqTC0JdkNtvngxK0ewuh5ORJlJMaVXZQjHQYkH

dEV4MqHnTLJ6KYqQCGF1sNkd/czruwvwAo3nNQaYHGcuvXqt/Dp87gc/t1ju4Nny0IeW+cVgkUGt

rq3i7hWjmhEGWQsYzG1r1tgspllS6Fl+F2YUl4HBsyhUAn3fPYl5Bgm69Xj6Oe9LE3SE3Xi843zW

NLjxxHg08DglBux+fQqq0GuCCPtCb5ZZMrlsMxRKtk4OiNLFAXqruegVkMI6K1MAZ3aAzFMIIQ1h

govtD5XhEXglSrpevvWef/WDgxV4gQfCHhq/8IY/P52xO8flu2hLJgkUYQEn9OAXwehRAAmaAQx7

T8AaXnACTaC8AA/cqsfvgUIcIPntwh/gDRn8AjsfiPBHxtqvffTlh2GK8ea3PfVkgR1DZn3I7K8q QjtXW4VdYY

+USvcmsehnCHEyi0xugd62AdJgQmNxGKp46YO2MqGQXPNZY4De65MwxgTkJdAXBvcM

lmY9t/dOxwsGgbYbP/5b31jLk9dFMMX41k/QhUz5EODenvo3ewFjgHpCDBCxvXPnOijuX3+CByKz

E2QJoAxmZVEhiSlYO/9kTabbZ9v07bzZSB1cIrtanBB+sADXLT2zYBITwHYQxya2xHOHIPxboA1a

nx6rlrWenQat0D0cSH8fCsIPkum1uGTtMw/Nnd4rwl/HnlKAOorV/xQ/bEliQW+CDBjGF4sElGK2

GLWlWsDgGgTz/gmChULH8xMeQTWGIRKSJpmY++fzyc5CWXwRoVNL7KgTS/gITDlKlP4pUrZHuM0g /KCWn

+lOIC0hwA5wQuB7vL4pRycWPn4W5uTG+6OMkDFDKeyfIRWEBJglzmLAzd4JanZ6FfbY2HMK

UO7kI8HqOlwnq3K21V6avE+wPc26pWjNNWDlbCqqpScuk4BAU4Z4gyED/6XhHYZzbNjW4JJSrqIo

vJVRrvWNtY/cRoOq32iLW8AGSSYkCj9bas2m9ut0JAFmkCMwOnxaivGBqzMpTEHEIPxcvoBS249b k2P2G9Xb

+GqPVvwGGkyjGRuiR2CPjT2nAO979JX88MWtKt6XIjrnI6JnUKYIZD6tY+X6qVglzjEo

Q0PY59eBt4QYgDOkQAj5AiCr1nSBVyWEqBSssep5VuE2R85wASKNG6YU6uorEmwaTAosEPUUpHoG

L3NGE25jaSJUYmtjUE7m9CosyiHA5n1AjDlaAR7F+n41JKlVdq7CHht7TgHqGBPXDbFwC+TxiSHt

CbUfgzBo9GT7CgSSk7+jFCIBeAyhwsr1RZxncKilXgEJzA0xNMukikM9rwG8L+rBiPpe3n4C+i3H

ID2iitcra5A5ugZ91BrXN7lVCXkSjDDMK7B6HIc1IuwA5g2y/KZQxuzKJ/i/C/vp0KlTBFuEL/jV

BpvKbm//HjwPY08qwOa9g/PlZn653r0s8IMCDmZgIPSMu+TeNiKsIFhdeThUmBQCXLKMJrNOZF5G

A0Xene1b2FLy+qK2QYA8Gm805V1ZmVi/a5rxaG1liPSIeBgpkyZLYKHN0JKg1fA/IUzClRSEkcUd

KTa1SnZPOaLHHjvEe9WKUEP+fK3qHThm0xhBAvBOPVd6sku7M6f3uY49qQBLKwdrh4cLXMhmSbAg

X6gQRdgflz6Kwo1GUwoNKrQMc/Y9K26WPwbC/F4IbHlkrQISlsaCdjm+Br51X3m8eceWrwD1ZQn2

yVga59LJLbSejccAWhJh/XlSgEgyQaYBf0TNC4BlenXhCrHoKsgaDflkdnBPohlldsoolaPINmr8

64X5WYM9OPakAtQxLl4Aa/8eYBNDBlfaLaYezHFrzY/Vyq1BLFhIKIUcAgQGSPeTDVqZ0Nt8gsiw

eAl3gFnWv5M9Vtlk+U6sv47qBYqwLrNtBe3GFjKrwu9r14akll0VRoJ8y+4iSU0OeU0QecCql6Te

QzE9aAYYyJpeJQzXTQaDOG7Q103Z1p6v0uY7GXtWAaoXKLj/ZK51jchxgFp/KZuom/Wts/aoF/sc

YIs4foc8GqjqZBOBP6wtYIkvecgqFfoaFCoZM9ULuPlc1/LLXvacF6EYHXjmQhUiO64WlHl9D3JW

Vq6FE10o7KnSomSUgl6zWP0wLdEoTJJ2JrxNiy308tGUqsEuhTh6nwg16SawqZb9nIE9PPasAtTx

/sIIlbt/vtKesh4vklto6vH4NcjNwNMENTCmWP2ZWFjFnlOw2LwPX9dWlMM9Awkj5O+r90FTLjDY w

+8jnVw5/9wTPpfOH9zADk9aiXNUYrXYzuKQF8SpkEvCSqtENV5iaCUTV8iZI63QNA5OvJ12fgB2

baSeIbRDdO6fIVCd0P6vHhzMvJ3hLMeeVoA25tIy1M5vaA/QmJn6MVlhnOBaebyciHLLr48cIjxi

AQ7Z2z7WFxrPhQZ6RXaaIZWElXgF5pmWZ7UwRR2fPP+S1cL9L4vFZgsvgbCVGwv7owmuSA7Etbki

7JHenBgwfbXoCKGWx4yC5gmSB8+VAfKgW40Kafyx9n88+MJUeN7O2PMKUKEQjG88WARrjSlFr5KM

mVh5RLBdFwmAmAGWIBedPs1C2UdWyMqwQQrflIZ1pondvLgJV064UIR/CWY8fvcXv24JUv51F2Lw Tmwq/IztA

+WJEEsTrB7H4QsBUhRwtNhKpysabQqeMNPePs1roJEM4EmztQOUjsFdMPa+B4DayvDV

awnGD5aX66DJMbHqgtW5fFoQEsMEioEsmJXWnboi8B6VtuE5lwKf9IeVQJCvBn/oNGF7t3qEqyv/

7Ot3zerR6KWn6zFM2Oqpa8Na+TPgfhNksJIG+dvX7ArBLXhWOQVjgaDzdtnKe6ZZ8L+zQuJpNqrw

71XWZ3LcFQpQx9InXnW1o/TdVJWgDfQJMuqegSYSWjxI+HoP1nSRbNKA15rLqnCDMebtwxArAIQM rXmNJpgv

+7pdtXptvd6XXj9WzvWqBrokHSBEUJWl4mtLZH9ThC/o/L1bc7HebUFrU26e8pgQe3VD

ybl/Ow/ez5eJxneN8NeBcJeN9z60Pkxd7V8Ph0Dml3IJMgustvRrGCEh9FoLAhiMyZ6vitifzDOA d7C2GEDhjniDrMoD

+Ah8/dedmUXQO+34Oz9z/WIR/EWfiC5Ur0Ed5fTlfmDg+IOXgO22UUjTFKIP idC9hiqIKthaLpb/T

+8i4a/jrlOAOt67uD7EET1WbvoRF34UClTSw1HgEWzegFtwMkH2nIHAHUJZ 39cVg7VA4gF0RSrvPLwbmH

+a8X1L15fKOX8AJKnVWoqgtxjxgBRCHKDUpWWDEeKEF52xZaUXAFoa Yd2iWSFIA+yym8fnqDt5N1l

+HXelAuh471vWl8rjOOuCGgS691otNUcITRDqXwmtm4MWrkUY1FMg

4Bdi9evrtUZ1zpDtuZPxfT93faFI4MXyctiqOAN2b+etc2+VuTHhh1C/I7mELhlEsoRXhJi+PyDl

/CEvF7ZnCe7ScdfEANuND/3GwaXCkR8uwntVcHgQWKt3JsfqbCnZcvezwM2aUV/4Pek1kVMgWKLx 5tEXWvjr

+L33v2R1bnTPUWqNxwh7c3XNYjMxgKoI9YvCWun0Rlnu1GhfY4Mc94euEO2/1aJNR+9m

4a/jrvYAcbznh9cXy7Or3mAIYuHJoU7bJoNgWDBWiK1jgFHqDQTmIBj338RhJR+YW96NFdhnMb7n

g9dLfNSdzYkWFZ+z8GNLyUquJCTOJCscyxiM6UGrKqW4YgvQ4+XuLO/F0uY7GS8aBdDxE4vrC+Nc

gkPEHzHcrjFA3YAVQKoe+EGzsijro8uKGjTaKGJzYZTz+ZWak7gLxvecuz4s1NApqCtBYoVGya/Q

oI1AIQ9myfB9h4HlaezQOrWOFi8ewdfxolMAHae5A/RCeXrHy4O8P9elQWFL8GsT6UOGeKPSjET5

cUhp9SN7AOY8l/G3z11fSIO0kLv0ANU10bq6+qNc94Twhwns1VtcK0K/Wl493t3bXdpL83hnOV60

CjA5qkKMihKUl/N1we0i3MM2ESbDRqoL7MG4/L736odX9ia8mdVY+OUbRwYJ5/MBqtc/n+prXpn1

mvRDvQovnVt7sQr8/tgf+2N/7I/9sT/2x/7YH/tjf+yP/bE/9sf+2B/7Y3/sj/2xP/bH/tgf+2N/ 7I/9sT/2x/7YH/vjxTn

+MzGW2Gjw5z/4AAAAAElFTkSuQmCC ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/png Content-Transfer-Encoding:

base64 Content-Location: chrome-extension://ofpnmcalabcbjgholdjcjblkibolbppb/static/monicaLogo.png

iVBORw0KGgoAAAANSUhEUgAAAIAAAACACAYAAADDPmHLAAAQAElEQVR4nOx9C5SdZXnu+33/3jOT

Gw5VwBQpEwRRwMUEKwTikUk5BQGRwQUH6S1BY616TkmOrtMe1jkrM65Te1YXSlJtV1sQJ9jWWlhl

tBWEqmyqFNQuGEAuEiCbckuoMpNMQpKZ+b+v3+V9v+/9/r33ZC57LmHmzfrzX/c/e//v8z7v5bv8

JXiDS/cG3W5WZhnrANAd/qjoqH+1rgLIIbPGpVzt7xND8AYWAW8Q8YrOuwBUh9TyTA260xzuAKf8

aYkFwIB5UENK6PvMIxsAKA28UYBxxALAK3y02yj7fA3QFa171sSAQgwoAd8EyCwgqnAEyhEFAKP0

DqnG1mthFe6W+SQVLbQBQ7n/SALDvAeAtXQJoxu0FpfD/FN6ItpQkfBPdMCAYZsBQ2W+g2HeAqB7

w2iXzOE6tPb2mm9q9zXMvuj6f5sOAztl94UWfSrT2/v7yhWYhzLvAND9W6Ndxoq2gLV2/qAFjP9t

ZwoQOlnVfgV2vggALsK6CDBA+FpLH8wjmTcAcIrXTPFW2LfTuC8Kx2dEdINDOuJM4H7xUne+DlrC phBVnave/r

+dH0CYcwB0X2kUn+ktILzihTF/TU/aHcCnic61risobk+RCbQex8J17TG7YTIB/33x

eLLPaMkfB3/WAVlWtVZXGCAMwBzKnAGgu1u3i9LIVqPX9fRNtNmxeiZrdztI/SHAqqfw6f4K3Xjf KZ4BQ2td

+5mpAC7cUPaZNLK3/+/mJlicEwBc1X3wOiVEj/nr7VHZSK0IBPpmQgpnMcCuC8L26bMT Fl3Y1JG

+o7I9AWnO9YdbT0RqgVsFqbb1f71tK8yyzCoAPtJ9oGMsF1/FyN5UXf1XSBWM+8mCF0j/

lDW6irrKP9wvKgZtOgLAfVzTMTzBFVzcL9wzBQEPCwvCgZxcIiowWrq2v3/22GDWAHD1JQdNSqd6

TD7frlFbmiuYqF9CLRtIvImEFBggCqygG8cKOs3gnGXXWDFDR6OleP1kXUEDBmPfe8gsm/tvm50g

ccYB0N2l21tbDmwxm5vcM2QWrQWzfsnOkUsQqUtw+wUQxPPIDkCBZMRIjU+HIgD0+MpWuKHYsUm4

guS7FNfFYzIc2Aqq1GvYYEbbHGYUABu6DnQclHm/0ciZmqJjZv1csTEOMNehksn/C5FeDwwILnB0

DKHDwwuWTrjgIOBRfSPFhwWVTp9VUNcleHchIEkVGBnx7+NE+KwAkt/NwA3heBW0Wtffv6QKMyQz

BoANXfs6RwHuNQoNgZ6iJ0GKLiodr7PbVvn2eQsZ3UAAg3tQIgBBBzYQNRRb9MSJ4kmhfFGNz2tF

MYJOsgPuSpKAkn8V5tbou3Flu+MyBUUAgRBX9PfPTLo4IwD4zfcfWC8g7wtM6YwDlYjX+HNR8W6f 6I

+5A3eeFC65G2BMIdLPAsQHHiQoSxeUDaniVXrOK90f1+g2RI2b0A3qBAUR7HsJEdwCATukucQE

BAjp/sbm/n9sfpbQdAD8zvuGt5gf2FNUPm17JmDUDzEO0IIpPjyMqHj3IHDfX6MdUwRgSfK1IrF+ DViwKVK

+glTxXOlW4SplAs1dQw0I4po2E+bh5IQbQkLyXAQFvAwIgj0D819P/z+Ve6GJ0lQAbDjP

KF9or3ymdI3clzyzohtAf84fiCazIH9PbCDZMTrPHhz3vXQ7t01BXLBynQKBlB62AYNFDXVdAv4B

wZQefAAUAADM77PfmGxzpYv0d7rjPm3u6b+ztWkgaBoAPmqUr7TusdupYYhapeM5VZcBGDuwfZAF

ZZPls3ggsZ6i6MKiClaf434OzOcXXALEtWYpIFWrk7QyoA6FdC/is6D9BAhShDgncQn4e4XfbhoI

mgKAj64Z3mKeTI/mdA9e2Z6A6wOBlB/0wtjAbqhgGdwNIAgydjxJDUXYPvZYgI4OCcuWAhx3jDD7

3g0sWypg2TJIfLjdfnUXwP59GvYPa9j9ioZXzbJ/L8DOp1VwB4K7AoBUyUXfT0Aoiiis0XXFYBbS

gDCjY8CAYEDwnemDYNoA2Lhmj1F+9PnkIhPKJwtJQCBSxQNaOMRYIHEJ+EA0CwjtgyFWWPV2CWe8

26xPkkbpRtnHgVN0s+S5HRqqP1Pw2L8p2GnWz/1MFUDAzL7IAIcDQdgW0eo5o+FvFsh+wv5u6fY3

9989vcBwWk/od88d7DYWfwe3cCu8QMaVHhSvG7gEOg+Q1gZYUKgRCMuOEnD2WqP0zgzOPtdadPOU

PRGxLGHB8KPv5/DYTxS8+rKKJ3WDDxUtnx9PAOF/o0iyHggMIJgbFJlJEe9p7YcpypSf2u+tGexQ

Ch7WrkFHBIdmKTYqPXUBzjiI3iGGSqHAJoB9DmKKiACwSl93sVH4+zI4vVPCfBILgu9/cwweNIDY

v6cBAsRhtkVhAeb3GSNYBoiuUAwJKdf1//PU6gRTAoBVvtGSKfJAh/fjqHBNLeHSqd0dx3bcem4g

YQao4yIQEKe9xyj9/ZlT/tLls2vpU5Hv9+fwra+NwHNPNggKBAsOaLvIDjweKPp/jAsQAPZ8Vei2

1f2VyZeNp/Q0P3nO4L3mW3QFhZtvpDjtY7Uj+ni/Hyxdx+tSd5CCwCr+yo+V4fSzMjgS5bEf5w4M

37tjDKNHrvwCOMjiAwNEYKTVTqAgMILAb1e+ee+SdTBJmTQAPrVmcIv57j1e6TrQuKdyzYJA+6U9

TSs8QNfqxC1oxiDgfuE7jcKv3FiG045QxRfl1Zc0fP3PRg0QRutf0MgdcEBQkYvcQQahCEZMoKXs

+VZlcpnBpADwqfe+2ill6eHEUily11CH2kVgCEhig+j/KWaw8k5j8R/e2PKGUXxRLCNsu37EAELV nmyUKRRdAs

+CuFtANjBB4br+ypIKTFAmDIBNXTvb8wNvetj88Q6i/qLPVoHaozWHY5D6ejpmzy9d IeEKo/gPfKQMC0G

+deuYWUYnBoRGMQEHQcZcQiaqQh408cDRE4oHJhxK6wPtW8zf7OCtHiKUc7zG Yw0GGzq0ZsSfpLt4U

+vnS/BHX1u6YJRv5UO/U4LPb2+DNReUak/WRsZxYQ1Svv1Cx+ql3bbrXJvs

rHULTFAmxACbDPVrmT2cpGjAonVO7XV8umYxgGviNceWmGj+8o2tcOHVC0fx9eTrXx5x8UFj0VDT

y6kQENYEhZktKBlX8MPDu4KJMUCW3RFiEcEtHbyVm1aTwEx2ccdSq6cN+wff8lYBf/DnSxe88q1c

899b4ObvLoVjj3dtvniUUwAEs3fPlCwLmUCzRiyd435um6f1jRP5+4cFwGfO+cV15o928O9C302I

yAbhG6GQe7DHJF5n/z/1rBL03LoMfuWU+VXImUs59njhXMJJ77QmzNPDNFX0sZdOXEKyTW7Ab3d2

n/v6JjiMjOsCNq15pUPqMhZ8kP4FBW+RlmIGIFhAGGmflvMubYFr/+8SWJTGcvMfj5gAcaTmeFI5

KLgCnx2ImC5m6A5KMCTKI6vGCwjHNcNMlXsAovV7iwakd1Sr5t/Fuwcf/EECrw9tbFtU/gRk4/9u gWs

+3VJz3JtT7UHuFkKX9hgctsNIy7gs0JABrPWXdHmn3U47w4rE6oFKwVrHTo+miE9FIYuISz/W

Ch80AFiUiYsPDuszAbFwOJb0ImKNRSXHBEOipTELNGSAksp6/B8Tob8aWXnR6n0KKGJfOZb6fXBR

+VMSGxw2YoJE0DdEg9RJRmYssV2NwwJ1GeAPjfUrKO1MrRzSBh2EYrB8nTYKWcR0/bdWuHLzIu1P

R/70+kOxhIza4rFA6GtIKSHGA+64Lww5FthrYoFKHRZowAAt63kaF3x/OB8jfP+3sc+/VsH/H39y

tqj8Jsjvf74Vzjgni8qn/B/pVjBFeZeLHCC06zSL/9qPUqW6LFAfADrfgPfCGoQO2+QKOM3bvJ+7

oTevlPCJP1kGi9Icuf5LbS5V1Kh4p1jsCueYWfpjgAqntWY5u9HgdV1dgzUzptUA4Po1r24w9+wI

/XSCv2eFCOCZAAR0Ekh+/89XwC+tXMzzmyXLVgi4/stLUNEQ+kB6ABAgdNhW6A7CeXDH21dA1lW8

d42WDJ2vT6p6kFJ/ZAMRg0B2zcUfa1tU/gzIqlMlbPyDNjdCWuPitzkT0KIZOIABQ1xXvG+iqR4T

/BmFdtntkNcD66AalA7M/0dWePPKDD6wcdHvz5Rc/ltlOOPsElM0WzIdF8cGKroHiSm51F1FN5AA

INeiOw32wAcZmsZdpklIZAd//NOG+hdlZmVTb1uicAhLtHzgoMBtyxbKrJe2yCQYTAAgQK4XPOCz

BymvDxXAyAzxc+Asf5H6Z16O+2UJv/mJ1mjpZq3sYhWcAe6b9L9kz4E/jou/XpzP7xc0Zukf59cF

buwx6ocEGOQarBxtqP+9l7bCosyOdF/TAkvfJLyiLRBKfgFck+U7pWcK2cIzhLm+6wPdgx10r2iy

OXTRDnXeIv/Ogz2eFVCHj4sWrX9WZdlyAd0faUXFg7f0wARk/QiIwAosPjAYontFrWXZersKBSVR Wya0yqZ8n

+SXFq1/TuSKq1phmWEBjfRPQHBMIJEJpA7WDyxgNNXBy+k+EQCK0T/WlwHSzh8U8PH9

CzcuhUWZfVluWOCKK1u9cm3PMgmR5kvgmMDvs0wAFyV0Z1e3zwYcAIz/7zItSO28j19QMvX144EA

A8FJZy326pkr+fCH29xoqSQjcGDArCD4fp4huHV7qQ2cwTsASC06Qxs+NS0BC/6oO5KIMYGV91za

ZgLARd8/V2JZYO3aFuwNzIpCjvJ1OO6UztyAH1nNAGAIoyuJ9K2v53V/OqlZJmCQ8J5F3z/nctGv

t8TycMatPJaNdegsmtQFuuznXb9kCfJE11sXWKmHhvcAVfo8BVD7grX8maT/vXv3wj333AMvvPCC

2z/99NPhtNNOg7e97W1wpAj/DUcddRSsWbPG/Y5myplnlGG5cQPD+9Ety6SzBrYWCkjnV7JAEGfa

y0o9nTvbbf7vB++IaOxChHoAB4XAkQknndUCMyEvvvgifOYzn4EHH3yw7vkrr7wSNm/ePO+B8JWv

fAW2bt3qQMDlwgsvhC1btjT1+593TgvcXTmYNAs73WEJWCAruH4CANR/sKNrw2C7zNraOj3ViyS6

F6wpkQahUBuAPXfa+c0HwBNPPAEXX3xxQ+Vbuf322+Hqq692185XufHGG+Fzn/tcjfKtWEaw37/e

ualK57tLcfIIiSzAB47GFM4zggMDgOHvDmnSv3bfzJt25IxFIB1cQwSIhreeUoJmirX8iT4Yutau

55tY5VvLH0/s9+7t7YVmyfve2+otn1r/JC/f+ho+B4g/bmp/Mu+UUnr6DyE+Wj71LqHMgHf4eOs7

Sq7820yZrFXYa62rmE9y2223HVb5JJYJmsUCy5cJeOtxInQOJXVKaRedTlJJ09C4c6LD9iHt8LdJ

R/NIxAOEdWwJbHbq98ADD0zJmq2rGM9dzKZY5X/2s5+d2AvQfAAAEABJREFU8PVW+c10A2ee3uKb

63HkVqRr4XsKU0DPR3Zl8kQLkhOptw9J8PWC6Z25g5WnNDf6t9YwVbn77rthrmWyyidpJgBOPrEU

+2wI7CsYCnY6ZAchubPXKDjaOfJQACpsaZ4KQswGrAtopkwnoJvrOMB+96ko36aFNq1tllg3AFK5 XC

+4dPABoX8JhoizjVGAIMWJJesC6B09FDdQ6sfbBTg82pbPn+pfM61osmKVb2OXqchVV10FzZST O8pR

+dR/Q2Jhnywe/bmLCfyp9pKJ/sM7+cKEL6Hw71nAA0q68X6WTtoXy79B+VMBoK0BbNp02HGb

k5LlS6J/9wGfjv7bzT4ac3kdi0MGAMK/XDlEibHq4zYlDfvSKjQdLplHDDAXYt3Oxz/+8Skr/xvf

+IZzAc2UZcv88HJn53ama0FFH8nmEcB6Lw0fAwMAChQklpB86OAvtNco1v2LSkOtKwQsVJlODYKU

PxNVzOV2VlSJGnIK1WwACaSdPGQsD5cEWTpA0uXT7ZuLsjDo00cBSsOClfmqfBIRe+9SFc9YOg7t

xcJAqAVgY1GJ0kWaUx8gBnvJNOh4LIOFaf3zXflOyO9jKgjYBhBr+cpOJef1jfFAKVQM3f9hGBG+

qDHCggaBCrEwKWDeKx9QX0T71AFEoGug9n13THp2MPt2yp8hLBO7OMD1ITBq90PLzdpEAXadmQtL dgygWfa

+ksNCkqlWKq3MVqvl7p/naO2+Z5DtDeyCPrvOYucQ4XoNKbxGD5lysBjCiaXcDL9u0dor

3X1Gx0X4DgQjwwuLBV566SWYqthS9dq1a10j0UzK/tdzR/Eg7TqPwZ5Tvu0anmO3ccW6kIEFgEGB

0DjCV7OeRCbtM4sbXm5uXALPAHY9vGsMFpKsWDH9EU+2kcgCwZaNZ0L2HaBmXh17/5RwXEAYM5Bj

V3IDhrI7XpWmKPA8xQtE9XZxM4y4RQXFu22z3vfyKCwkOffcc5tC5daN2LKxbcVsdgn72RcPoe+H

YPXO8q3Fl3O3KKN0ZUFgi4ZlDwRplDvkppsVGsJso0Hh3urdvjtmgaRgz9MjsJDEFm1uuummpvlz

26nFdnyxvYaaJbt/MZYoX5Py0eoVWr23fAMEc16VDQOYyLEa3sCCbiDDAJDiAM8ECASzvWfHIVho

YhtubDTfrBq+rSLaXkPN6tPw3IsjIEpM8c7n50HhSPlW6aBblFtUNva8zQeqEinfMYAg68cAkJhA

2C5EfnuvYYDRYQULTSwD3HDDDU1ng2YwwaPPHvBzBlgAmCBQlHXw97o05mjfugBL/RYEjhHk2IDM

QA4465esGxl4l2CPOSCIAhjMMrSjeW6g2XXxmRbbsfP+++93nVObIRPtRdRIHt1hlZ97y5fe2lWJ /L9G5VvLN

+sWv7YLLDVZQLlltEr9/EVgAh2CQkmuQFCq6IHyyn37oVkyHWtqZpv6ZMW26FkgTNct

WHcwnaDwXx/b56J8l/cHyx/DQM/SPVO+YwC7bZS5HAbktZVVJheEqksbJS4QO5NKUjwpH4/vaiIA

LrroIpiqTOezzRByC1/4whemBeRpAeDx/T63J+U7f+8HiuqWzFN+Wfm1BYJnhoHKtauGfKOghEew

OugVL1HhgjUgAeBM4d41HDS1gJ8/dBCaIXbAhF0mK/aBT+VzMyF2vMJdd901ZbcwVSZ77uVDsHuP

cccu6MOAz7KBS/UUAsIuprxvm/5K6BJK+nn7eWzY1xWBtWEIStdsaBh2LQbqU+DP/eKhA9AssRY0

mVjAXmuj8vkk9juRW5gMG1gQTzUO+ocfDKL1+wkjfLnWOO6SidTKJTdS2G3btQOFIABU7OdpbOAA

H/0Ru4lDaDQII4TDomHnNyb9lrKGMplGE1L+fB0dZL+XBcFE3IL9Lfa6qcqj1QPo942O0Nqd0stY

8i1LdAfSAMQuwp0XZeneM0hdewwAYCgMLbJrfDkhTTvquhrEvmTuv9H9Gl64cxiaJRPJta21WKqd y

+BvomLdwni/Z7oFpnse2gu7hsd8m65L97yioZzZuYCM8jOX3nnli8gCJTH04FWrKvYeoXH/r9e9 cK

+dIi7DFj8JcWLIoHPFJ4fSDg1vWd0G5315JTRbbFBk+909/vjjbv+EE05w6deRljKS2N9zyy23

uN9jFW4BbIExnd/z21/cCbv2jqClm6i/lPkI39G8Dfz8vmUHVfLRvHKjg2Tl3y47aZ29R9Dnrb/2

/KYMxI0SARCZns8QqmmQcBg4amXtn650QFiU2ZN7Bobhhv5drqQLhuYVFXustZc99VsgqJK3fIUF

HTdhBMC1D116cp+9T+jdmSvVH97aLe2r2z3122lHlfTIscfpnFtw+8m+5sUCizIx+Vrl5175lO5Z

K3c0Lz3tk7LdvEHg93GtZFah+wQAmHpA1fj8ileq7zLkQCAJCAJyVLrGawgcrz5yCHbcPnf98xea

3DOwB14ZPoTRvlU0KtxG+y7IyzwY6Di28uHkEAMDFxtdoyT9u3MB91nyDxaOSy49uqyycwm4+OPm

M255fPseGN23gHuMzpLs3jMGt/7gNbR6if7dbre4GMAdy2QI/qLl4yLUdn6/BABqZGxrPYqP+1B7

Tvrl4OsaHrt10RXMtGz/l9fglX0j3tIdvVull111z88VKIJL0GGmMHThNpArjfXz+yUAsGVhc6RC

tK/J/xP929FBbklBoBEET/3DPtj9yMJrKp4tufvRYbjblH3BKNwr3ejFBXjCAcEFeSXUhzmeZxS/

0RRxUBlY964qv2fNEB8TDG4jn+8DQR4H0JKyRG7dBrqOB254DfbvXlidRmdDLPVvv3/QWXQuASt8

WQj2nPWXaFvEeQKzOFuIEnp78b61Y7yW5hWj9KFU4ZgBiDQLCEyBgaFd9hrl329AsCjNlf//7d2G +pWn/bIP

+BTGAX6+4MxPEJkhOGiewGD9qvrYunf0Fe9bA4Br+1cNKWFZABUu4jt/VUgT0T0AsQEE hrAgeOWxEfjxX

+yBRWmObP/hEDz80ij6fOlzfdzOM5/rqzBLOIKA3iBiQJK72UFhe7171x3lKVtM

MAjgIjpfAvYNASEuAAAeJ1BaGM6Z5fH+fWZpXpPxQpXt9w/DVx/Yi1U+o/Cyr+nnWN71dQDAqeLJ

+jUWfyIDjCnVV+/+dQEQWADigFCNDUQaR5cSMPxLI9l2YAKAH/3VHtjx3ea1GC40ueun++CWBwZD

gUdRg05JYOEHAz18P4BCyrfrXCrHArnP/fueKgR/JA3HeZcMCyQNRDhczG3TmEHeSASMLaQHimWD

+24chKe/twiCycpdP30d/vg7Q8aaM1fYsRbvmDaLFO8DP6PkEm1DeHlEyPsz1wO4t9HfaQgAxwKg

tsWmIJ0onLaBNREnoHDbHgiVLw7BzxZBMGGxyv/8XUb5MnNW7+gcZwEPbwjB+n6e8beGKO/vGQMY

0PQ9dW5967cy7kwPJRcL6CGNQ8iDYtl2crxG+VEqX9wDj/a/Dosyvtz10wNG+Xt9gFeKFE+vgsmp

STd0/46vi1HsxRG+4UdXzZHe8f7euACwLGAiyV4Bk5Wk0RjotfL337QXfvI3i4FhI7nl/v3wR075

qPQSexMIvgxCo6U76g/WD/HlEPSOIDtOUKtt41m/lQnp9uZLXnrYrDrrfbhY/cdZhgpsEVNJGyOc esES+C

+/uwJal00eWm9EGT6k4UvfG4ZvP3HQKVAw5QL59zDWz5wvW2bAgR52LAAChQJAfJNY9ek1 71p1uL89ocl

+TBHB9XTk8wm748XrAKARpgIwzMaTJh742//xC1M0WniDS4qyY/cYfPSrrznlA3sL

mPbj9NGPx6IOoH9314n0jWCKBoba6/P8ion8/Qmb4M2XvLBVgHRvntTswzrhAVEnTojHFKaMCrMK

q/5zfmM5nPcbC/O1M3//kwNwy7/uh+FR5ZVbQmUHy0ZlY/AHONrX9fJxQFEhPgD2OWOw255dc9qE piGbMAC

+2r2zXY20Pmxu31GP+gEIDFBD/woiGBQGigqPW1CsODaDj3y+Hd503MKYfWyHKZd/6bvD

8JCp7lG1LrzMIdPpW75IqfQ6uLIOyo+pXnxBlGGN6mhr6+rq6lUTapqdlBO+6YMvdAkl7613jvt+ vs

+VH8rHAKHxKGyb5Vcvb4OzL1vyhgWC9fW3/9hY/Q/3o8Ig0DpXpkKKd8rHXB4w5/dsoIKbUPhK

GIUTOwg9tnrH2jMGJvqdJh2F3XTJSz3mQ1uKxw8b/DEA2G3bXpgAQPgZyI4ybPB+4xI6L5iZF1LM

hewzir/tRwfgNqN8ont6q6dgyneRe4m/AVSFl0HFjCC6BB1eF4vWD7r3mbXv6pnMd5tSGH7zpS/f a5x5F

+3z4K8+AHBdAIAW3vJzAocW4RoLhK5rlsBZRzAQ9h00Fv+jg0bxr8PwiE6sXqPlhkDPHY+W

nbz3r1RwCwE0oZ3fbleeOe+d62CSMiUA2HggN/GA+bMddp8rnPbTABBdgYCE8r2yi9vp+k3HZbDq

jBL812va4Ohjj4y00Sr+NqP4243VD48oXyCTqcUC+ntgr3nlqV586SMdA98JlN4NyABgPls1aeO6

6mFy/noy5Sf6V5e82ClB2HigvZ7/r4n+gVt/DAbzEAuIEB9oIGaIbsIeO8kA4VcvKMN7f23+vavQ

Kn2gOuYsfuD5EWwpReukmbtEDPRIka6jBrPs4PNltHJ6FawuBIr4evih0Vytrq6bvPKtTMukbr7s

37t1nt0B40b/9ehfhFSQ6D9n8UJwC4C9kQvn2pYLePu7S/Dus0twyhkZvHmOmMEq/YdPjsIjRvE/

eOoQ7BtBJdtfZtvp2Xx9/M2dgO/1iy941qzhhgeGWOSRHgRKqti7FwGlFFzx7LpT+2GKMu0nd9Ol

L28yv/fG1PoL/l9wy4aa6L8R/afXsXtIX08gIB1/knRAeMfpGfzKKglvOWZmALFrUMEzu3J4dOcY

PFwddds0dMpbvKd7+51EULz37e58Fqk/vN8v4yAo0H8JgcPa9RVO0OCYAfLeZ86fXNBXlKY8KZsZ

GMVsaRT8RQbgSq2v5MAQkAKF2EDp6FJy+hx2SMld51UNS1ZIOGGVgBM6JBxjwGCXpabsfMxb/JCm

495c/2fvHvS/wCp612vKWLiCZ18x20M5PGPW+w4pN3ReYcsoIM3HtQ5TsEaLR1dAffMEhIAvUX4I

CNHKWZlXh5Y9CK7D/Ot9et2pPTBNaZqp/KUBAQQQRGXGwk+D4I+OswIRp3+KBXRyHfssKp46qzpQ SN

+YkgvclhBfrY5reqeuL59q6jQZpluNU66Cp3Tpp9UOg2PJn4N/I5fCKXRr39ypo+VLCIouNt4Q

KOI7fxEQ7Bqif/Mze59ad0oPNEGaVnH5xJ3H9xhM99YUg3QhTawHOUHXi5oUMskwBF5caHYON9EQ

X3ahC/eIHwjbGtJ2DSHY9QEICA4g6450rBn9k9UDt3wZUz33h+7oGK4AAAXbSURBVFzeH4M/LWJG

EOb4C5M8QwQRMo72p5umfIAmAsDKJw0IzJfs9XsiWVFcUFRMvZQx9DWEYmCJTKDjPp++3p1HJbu3

oLqBrcIPbNXYlKV9o5a7SvPvp2uUn9C85BSvE6XrgtJjsOct292ffLjA4C24hwiy8Ap4gQ09jIXc O/

+s8i9onvIBmgwAK5+8c2WP+QGba4NCXEQdhRev0fEcFO7jRSTWz68jFAgo/FHcD1eyY7GdE1+N

yybH0KLo73VwDzp5zx7EiF9ASPmiNUNkC7wWCnFCZAUIbBCAqMS1T/16c5UPMAMAsPKpb//yVnPr

1cajVcez6lSpXgJLMyunY9Doc+6VNgxyIjJBcEGkcKpIFRFZR1Llx7XmQSB1wACd+Hj+fp7oGiBx H+H

+LK2L1O/PmZhkyDy91U9deHIfzIDMCACsfPrOYwdypdeZ51K1+1ozXx1WqT/nlcTEzydsEa0/

AYcAcuRB2SJRegoKERDJkAYQXAH/qlCwfEioHoKfx+g8+niZXqvILQieImp0L7Htn36PUuoRpUqr

n7jwHRNu3JmszBgArGz+zsqqamldbX7wtkTRoSs51LgBqHNMixQc5OsDOKzLUJ66NSpaMMsXjDZ0

rEgBy0ch3BwQbkWvwhmAwBBomgWFJYiWLyC6BJfCxWNQcBEJY3gwbzs4Uu56ig3lngkRMEvyJ5e8

ugGkuNGkae28L0DSOgh+bEGOx3iKGGoBUEwdqZ0hpoUhzZQ+FSRLq00Dqf4OIS9PWtec0lSMxgsU

TQqkIFFTtM8tnKg99O5BF8Vy/6Rp2JR2c6l6n7zklOlNHzpBmVEG4PK/7jy2Ly+NrTa/vmL3Y+DG A75iewJuF

+MBdB3J9RqS+yQfIOsPo1YsYwC9J9ulEpQthODf00e8Jz8RrD8qP6R6glM7BKAo5vND

6sfTPs86lbFMrp4t5QPMIgNw+X+X/ccmBfI6o4OOw5Z+sRgUGVuk1i8gMANvU8iDrkXwrY5ZeHEF X6MS+

+BpX48PlqliQUayII38eBatGViUT029aWrImUOH9gDlgTBkB288/qG3z5riSWaNAbj8n388

ZqvK5DrzeLcXjBR4Mcj799pzFBOEWACDPa0p6o85vz8hwvtznREiukJ9QAl2TGCAyIpEjBlCfsnT

v/BndKwXsEKSjxd0YvkxgIQ+g4VVc6F8gDliAC493YOdIzncYR5SB80zkFi7qI0HdOE6XWxPEHHb

WhqVh0NpNrTC4RobXYB3vpC8Nw6z4NAJw59z065lmtX8IxCUTCuCvJZvJ+IwhareAZyvb65kzgFA

8oeXDW4wLmCLUVxHSv9pu0FUbr3+Bhgwah9MKrS+4AoEUz4CIMfI3SmHvVApKJr1xgHe+VISUKi8

C7XVwCx1A+j/KwYwc654knkDAJL/2T24QetsvaHdLt5MrCFtNua9jZUQkPQ/sLGAjiOVncLdcQQE

+mzFrTrMpqGB98z1DEAtd9g5U/KYIWYMitUFQs4fgGCCX5H3DlwzPxRPMu8AQLKpe7hrDJQBg1zv

0kSe8hXYIASBkKaWfh4D317Am2kV+mR7TwrwgIOgpEOK6DtpqqDsYts9FHvqJAwAQ7avnhJj2wZ+

e34pnmTeAoDk97oHOwDKXUYF1xmldga61zEY5H0GNE1fA5B0IoGQywM2xgA2AUNiyUmXK8wKqLdO

GKolVcgg4jHNO3hWtFDfHCuN9g1cu2peT5027wHAZUP3gQ4JebcW2eXmMXcpKHY6jeswAgmPUc7u

tqlII1jgRgUbFvDRuDx69144JlXaa9e7i4oBwH0mnuh78JMrq3CEyBEFAC4WDMZFdIKQ3QYAZxpF

d1IlUDM2iGkjhIactMVO+LSNVewoJgAZe+pCGIwRWKGqy7piMoz7Wg4N91d65relN5IjFgBF6e7W

7W2lg53WTRiGON+4iHajKusy2u35BAgUkWMe71mBp3cRBMbah+xwK7M9oFtGHzHUXzVWXqn0HP2G

mBXzPwEAAP//76VoAQAAAAZJREFUAwCq4BBDmEFA1AAAAABJRU5ErkJggg== ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/png Content-Transfer-Encoding:

base64 Content-Location: chrome-

extension://ofpnmcalabcbjgholdjcjblkibolbppb/static/global/src/static/monicaLogo.png

iVBORw0KGgoAAAANSUhEUgAAAIAAAACACAYAAADDPmHLAAAACXBIWXMAAAsTAAALEwEAmpwY

AAAA AXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAACXdSURBVHgB7X0LlF1Vmea/97m3qvLCotVgGmkq

KKKAiwq2EIgjVc00CIgULhikXwka21ZnmmR0TfewZlaqXNP2rF4oSev0mmkQE5xpx4bVlHYLQqtc

WmlQe0EB8pAAuTSvJEupSiohSVWdvXs//n/vf597b6Uetx6h6s86Oc976t7zf//3P/bjCHiDS88G

3W5WZhnrANAd/qjoqH+1rgLIIbPGpVzt3y6G4A0sAt4g4hWddwGoDqnlWRp0pzncAU750xILgAHz

oIaU0PebRzYAUBp4owDjmAWAV/hoj1H2BRqgK1r3rIkBhRhQAr4NkFlAVOEYlGMKAEbpHVKNrdfC

Ktwt80kqWmgDhnL/sQSGeQ8Aa+kSRjdoLa6A+af0RLShIuGf6IABwzYDhsp8B8O8BUDPhtEumcP1

aO3tNd/U7muYfdH1/zYdBnbK7gsttqtM7+jfXq7APJR5B4Ce3xvtMla0Bay18wctYPxvO1OA0Mmq

9iuw80UAcBHWRYABwjdatsM8knkDAKd4zRRvhX07jfuicHxGRDc4pCPOBO4XL3Xn66AlbApR1bnq 6/+b

+QGEOQdAz1VG8ZneAsIrXhjz1/Sk3QF8muhc67qC4vYUmUDrcSxc1x6zGyYT8N8Xjyf7jJb8

cfBnHZBlVWt1pQHCAMyhzBkAenp0uyiNbDV6XU/fRJsdq2eydreD1B8CrHoKn+6v0I33neIZMLTW tZ

+ZCuDCDeV2k0b29f//uQkW5wQAV/ccvl4J0Wv+entUNlIrAoG+mZDCWQyw64KwffrshEUXNnWk

76hsT0Cac/3R1hORWuBWQapt/d9s2wqzLLMKgI/1HOoYy8XXMbI3VVf/FVIF436y4AXSP2WNrqKu

8o/2i4pBm44AcB/XdAxPcAUX9wv3TEHAw8KCcCAnl4gKjJau6++fPTaYNQBcc+lhk9KpXpPPt2vU

luYKJuqXUMsGEm8iIQUGiAIr6Maxgk4zOGfZNVbM0NFoKV4/WVfQgMHY9x4yy+b+22cnSJxxAPR0

6fbWlkNbzOYm9wyZRWvBrF+yc+QSROoS3H4BBPE8sgNQIBkxUuPToQgAPb6yFW4odmwSriD5LsV1

8ZgMB7aCKvUZNpjRNocZBcCGrkMdh2XebzRylqbomFk/V2yMA8x1qGTy/0Kk1wMDggscHUPo8PCC

pRMuOAh4VN9I8WFBpdNnFdR1Cd5dCEhSBUZG/Ps4ET4rgOR3M3BDOF4Frbr7+5dUYYZkxgCwoetA

5yjAfUahIdBT9CRI0UWl43V22yrfPm8hoxsIYHAPSgQg6MAGooZii544UTwplC+q8XmtKEbQSXbA

XUkSUPKvwtwafTeubHdcpqAIIBDiyv7+mUkXZwQAv/vBQ+sF5NsDUzrjQCXiNf5cVLzbJ/pj7sCd

J4VL7gYYU4j0swDxgQcJytIFZUOqeJWe80r3xzW6DVHjJnSDOkFBBPteQgS3QMAOaS4xAQFCur+x

uf/vm58lNB0Af/CB4S3mB/YWlU/bngkY9UOMA7Rgig8PIyrePQjc99doxxQBWJJ8rUisXwMWbIqU

ryBVPFe6VbhKmUBz11ADgrimzYR5ODnhhpCQPBdBAS8DgmDPwPzX2/8P5T5oojQVABvON8oX2iuf

KV0j9yXPrOgG0J/zB6LJLMjfExtIdozOswfHfS/dzm1TEBesXKdAIKWHbcBgUUNdl4B/QDClBx8A

BQAA8/vsNybbXOki/Z3uuE+be/vvam0aCJoGgI8b5Sute+12ahiiVul4TtVlAMYObB9kQdlk+Swe

SKynKLqwqILV57ifA/P5BZcAca1ZCkjV6iStDKhDId2L+CxoPwGCFCHOSVwC/l7ht5sGgqYA4ONr

h7eYJ9OrOd2DV7Yn4PpAIOUHvTA2sBsqWAZ3AwiCjB1PUkMRtleuBOjokLBsKcAJbxVm37uBZUsF LFsGiQ

+323t3Axw8oOHgsIY9r2rYa5aD+wF2PaOCOxDcFQCkSi76fgJCUURhja4rBrOQBoQZHQMG

BAOC700fBNMGwMa1+4zyo88nF5lQPllIAgKRKh7QwiHGAolLwAeiWUBoHwyxwup3SDjzvWZ9ijRK N8o

+AZyimyXP79RQ/YWCx/9FwS6zfv4XqgACZvZFBjgaCMK2iFbPGQ1/s0D2E/Z3S7e/uf+e6QWG 03pCf3jeYI+x

+Du5hVvhBTKu9KB43cAl0HmAtDbAgkKNQFh2nIBz1hmld2ZwznnWopun7ImIZQkL

hp/8MIfHf6Zg7ysqntQNPlS0fH48AYT/jSLJeiAwgGBuUGQmRby3tR+mKFN+an+0drBDKXhEuwYd

ERyapdio9NQFOOMgeocYKoUCmwD2OYgpIgLAKr37EqPwD2RwRqeE+SQWBD/89hg8ZABxcF8DBIij

bIvCAszvM0awDBBdoRgSUnb3/+PU6gRTAoBVvtGSKfJAh/fjqHBNLeHSqd0dx3bcem4gYQao4yIQ

EKe/zyj9g5lT/tLls2vpU5Ef9ufwnW+MwPNPNQgKBAsOaLvIDjweKPp/jAsQAPZ8Vei2Nf2VyZeN

p/Q0P33u4H3mW3QFhZtvpDjtY7Uj+ni/Hyxdx+tSd5CCwCr+qk+U4YyzMzgW5fGf5g4MP7hzDKNH

rvwCOMjiAwNEYKTVTqAgMILAb1e+fd+SbpikTBoAn1k7uMV8916vdB1o3FO5ZkGg/dKephUeoGt1

4hY0YxBwv/DdRuFXbSzD6ceo4ouy92UN3/xfowYIo/UvaOQOOCCoyEXuIINQBCMm0FL2fqcyucxg

UgD4zPv3dkpZeiSxVIrcNdShdhEYApLYIPp/ihmsvNtY/Ec3trxhFF8UywjbbhgxgFC1JxtlCkWX

wLMg7haQDUxQ2N1fWVKBCcqEAbCpa1d7fuhNj5g/3kHUX/TZKlB7tOZwDFJfT8fs+aUrJFxpFP+h j5VhIch3bhszy

+jEgNAoJuAgyJhLyERVyMMmHjh+QvHAhENpfah9i/mbHbzVQ4Ryjtd4rMFgQ4fW

jPiTdBdvav18Cf7sG0sXjPKtfOQPSvDFHW2w9sJS7cnayDgurEHKt1/oWL2023ada5OdtW6BCcqE

GGCToX4ts0eSFA1YtM6pvY5P1ywGcE285tgSE81fsbEVLrpm4Si+nnzzqyMuPmgsGmp6ORUCwpqg

MLMFJeMKfnx0VzAxBsiyO0MsIrilg7dy02oSmMku7lhq9bRh/+Bb3ibgT/5q6YJXvpVr/2ML3PL9

pbDyRNfmi0c5BUAwe/dMybKQCTRrxNI57ue2eVrfNJG/f1QAfO7cX11v/mgH/y703YSIbBC+EQq5

B3tM4nX2/9POLkHvbcvgN06dX4WcuZSVJwrnEk55tzVhnh6mqaKPvXTiEpJtcgN+u7PnvNc3wVFk

XBewae2rHVKXseCD9C8oeIu0FDMAwQLCSPu0nH9ZC1z335fAojSWW/58xASIIzXHk8pBwRX47EDE

dDFDd1CCIVEeWT1eQDiuGWaq3AsQrd9bNCC9o1o1/y7ePfjgDxJ4fWRj26LyJyAb/2sLXPvZlprj 3pxqD3K3ELq0x

+CwHUZaxmWBhgxgrb+ky7vsdtoZViRWD1QK1jp2ejRFfCoKWURc9olW+LABwKJM

XHxwWJ8JiIXDsaQXEWssKjkmGBItjVmgIQOUVNbr/5gI/dXIyotW71NAEfvKsdTvw4vKn5LY4LAR

EySCviEapE4yMmOJ7WocFqjLAH9qrF9BaVdq5ZA26CAUg+XrtFHIIqbrP7TCVZsXaX868pc3HIkl

ZNQWjwVCX0NKCTEecMd9YcixwH4TC1TqsEADBmhZz9O44PvD+Rjh+7+Nff61Cv7/xHdmi8pvgvzx

F1vhzHOzqHzK/5FuBVOUd7nIAUK7TrP4r/04VarLAvUBoPMNeC+sQeiwTa6A07zN+7kbevMqCZ/6

i2WwKM2RG77S5lJFjYp3isWucI6ZpT8GqHBaa5azGw1e39U1WDNjWg0Abli7d4O5Z0fopxP8PStE

AM8EIKCTQPLHf7UCfm3VYp7fLFm2QsANX12CiobQB9IDgAChw7ZCdxDOgzvevgKyruK9a7Rk6Hx9

UtWDlPojG4gYBLJrLvlE26LyZ0BWnyZh45+0uRHSGhe/zZmAFs3AAQwY4vrifRNN9Zrgzyi0y26H

vB5YB9WgdGD+P7LCm1dl8KGNi35/puSK3yvDmeeUmKLZkum4ODZQ0T1ITMml7iq6gQQAuRY9abAH

PsjQNO4yTUIiO/jjnzXUvygzK5v62hKFQ1ii5QMHBW5btlBmvbRFJsFgAgABcr3gAZ89SHl9qABG

ZoifA2f5i9Q/83LCr0v43U+1Rks3a2UXq+AMcN+k/yV7DvxxXPz14gJ+v6AxS/84vy5wY49RPyTA INdg5XhD/e

+/rBUWZXak59oWWPom4RVtgVDyC+CaLN8pPVPIFp4hzPVdH+oZ7KB7RZPNoYt2qPMW

+Xce7PGsgDp8XLxo/bMqy5YL6PlYKyoevKUHJiDrR0AEVmDxgcEQ3StqLcvW21UoKInaMqFVNuX7 JL

+2aP1zIlde3QrLDAtopH8CgmMCiUwgdbB+YAGjqQ5eQfeJAFCM/rG+DJB2/qCAj+9ftHEpLMrs

y3LDAlde1eqVa3uWSYg0XwLHBH6fZQK4KKE7u3p8NuAAYPx/l2lBaud9/IKSqa8fDwQYCE45e7FX

z1zJRz/a5kZLJRmBAwNmBcH38wzBrdtLbeAM3gFAatEZ2vCpaQlY8EfdkUSMCay877I2EwAu+v65

EssC69a1YG9gVhRylK/Dcad05gb8yGoGAEMYXUmkb309r/vTSc0yAYOE9y36/jmXi3+7JZaHM27l

sWysQ2fRpC7QZT/v+iVLkCe73rrASj00vAeo0ucpgNoXrOXPJP3v378f7r33XnjxxRfd/hlnnAGn

n346vP3tb4djRfhvOO6442Dt2rXudzRTzjqzDMuNGxg+iG5ZJp01sLVQQDq/kgWCOMteVurt3NVu

838/eEdEYxci1AM4KASOTDjl7BaYCXnppZfgc5/7HDz00EN1z1911VWwefPmeQ+Er33ta7B161YH

Ai4XXXQRbNmypanf//xzW+CeyuGkWdjpDkvAAlnB9RMAoP6DHV0bBttl1tbW6aleJNG9YE2JNAiF

2gDsudMvaD4AnnzySbjkkksaKt/KHXfcAddcc427dr7KTTfdBF/4whdqlG/FMoL9/vXOTVU631uK k0dIZAE

+cDSmcJ4RHBgADH93SFCq3Tfzph05YxFIB9cQAaLhbaeWoJliLX+iD4autev5Jlb51vLH

E/u9+/r6oFnygfe3esun1j/Jy7e+hs8B4o+b2p/MO6WU0ClCZEcZgA69Sygz4B0+3vaukiv/NlMm

axX2Wusq5pPcfvvtR1U+iWWCZrHA8mUC3naCCJ1DSZ1Gt2bR6SSVNA2NOyc6LFY6/G3S0TwS8QBh

HVsCm536Pfjgg1OyZusqxnMXsylW+Z///OcnfL1VfjPdwFlntPjmehy5Fela+J7CFNDzkV2ZPNmA

RJxMvX1Igq8XTO/MHaw6tbnRv7WGqco999wDcy2TVT5JMwHwzpNLsc+GwL6CoWCnQ3YQkjt7jYLj

nSMPBaDCluapIMRswLqAZsp0Arq5jgPsd5+K8m1aaNPaZol1AyCVy/WCSwcfEPqXYIg42xgFCFKc

XDI7HfSOHoobKPXj7QIcHm3L50/1r5lWNFmxyrexy1Tk6quvhmbKOzvKUfnUf0NiYZ8sHv25iwn8

qfaSif7DO/nChC+h8O9ZwANKuvF+lk7aF8u/QflTAaCtAWzadNRxm5OS5Uuif/cBn47+280+GnN5

HYtDBgDCv1w5RImx6uM2JQ370io0HS6ZRwwwF2Ldzic/+ckpK/9b3/qWcwHNlGXL/PByZ+d2pmtB

RR/J5hHAei8NHwMDAAoUJJaQfOjgL7TXKNb9i0pDrSsELFSZTg2ClD8TVczldlZUiRpyCtVsAAmk nTxkLA

+XBFk6QNLl0+2bi7Iw6NNHAUrDgpX5qnwSEXvvUhXPWDoO7cXCQKgFYGNRidJFmlMfIAZ7

yTToeCyDhWn98135TsjvYyoI2AYQa/nKTiXn9Y3xQClUDN3/YRgRvqgxwoIGgQqxMClg3isfUF9E

+9QBRKBroPZ9d0x6djD7dsqfISwTuzjA9SEwavdDy83aRAF2nZkLS3YMoFn2v5rDQpKpViqtzFar

5Z5f5mjtvmeQ7Q3sgj67zmLnEOF6DSm8Rg+ZcrAYwoml3Ay/btHaK919RsdF+A4EI8MLiwVefvll

mKrYUvW6detcI9FMysHXc0fxIO06j8GeU77tGp5jt3HFupCBBYBBgdA4wleznkQm7TOLG15ublwC

zwB2Pbx7DBaSrFgx/RFPtpHIAsGWjWdCDhyiZl4de/+UcFxAGDOQY1dyA4ayO16VpijwAsULRPV2

cTOMuEUFxbttsz7wyigsJDnvvPOaQuXWjdiysW3FbHYJ+7mXjqDvh2D1zvKtxZdztyijdGVBYIuG

ZQ8EaZQ75KabFRrCbKNB4d7q3b47ZoGkYN8zI7CQxBZtbr755qb5c9upxXZ8sb2GmiV7fjWWKF+T

8tHqFVq9t3wDhMyCwTCAiRyr4Q0s6AYyDAApDvBMgEAw2/t2HoGFJrbhxkbzzarh2yqi7TXUrD4N

z780AqLEFO98fh4UjpRvWMBYfotyiyqNvWDzgapEyncMIMj6MQAkJhC2C5Hf3m8YYHRYwUITywA3

3nhj09mgGUzw2HOH/JwBFgAmCBRlHfy9Lo052rcuwFK/BYFjBDk2IDOQA876JetGBt4l2GMOCKIA

BrMM7WyeG2h2XXymxXbsfOCBB1zn1GbIRHsRNZLHdlrl597ypbd2VSL/r1H51vLNusWv7QJLTRZQ

bhmtUj9/EZhAh6BQkisQlCp6oLx6/0FolkzHmprZpj5ZsS16FgjTdQvWHUwnKPznxw+4KN/l/cHy

xzDQs3TPlO8YwG4bZS6HAXldZbXJBaHq0kaJC8TOpJIUT8rH47ubCICLL74YpirT+WwzhNzCl770

pWkBeVoAeOKgz+1J+c7f+4GiuiXzlG9Zwa4tEDwzDFSuWz3kGwUlPIrVQa94iQoXrAEJAGcK967h

sKkF/PLhw9AMsQMm7DJZsQ98Kp+bCbHjFe6+++4pu4WpMtnzrxyBPfuMO3ZBHwZ8lg1cqqcQEHYx

5X3b9FfyLiEv6Rfs57FhX1cE1oYhKF2zoWHYtRioT4E/96uHD0GzxFrQZGIBe62NyueT2O9EbmEy

bGBBPNU46O9+NIjW7yeM8OVa47hLJlIrl9xIYbdt1w4Uwq9LumI/T2MDB/joj9hNHEKjQRghHBYN

u7416beUNZTJNJqQ8ufr6CD7vSwIJuIW7G+x101VHqseQr9vdITW7pRexpJvWaI7kAYgdhHuvChL

955B6tpjAABDYWiRXePLCWnaUdfVIPYlc/+NHtTw4l3D0CyZSK5trcVS7VwGfxMV6xbG+z3TLTDd +/B

+2D085tt0XbrnFQ3lzM4FZJSfufTOK19EFiiJoYeuXl2x9wiN+/+3+8X77BRxGbb4SYgTQwad

Kz45lHZoeMuaNjj/q6ug2WKDItvv7oknnnD7J510kku/jrWUkcT+nltvvdX9HqtwC2ALjOn8nt//

8i7YvX8ELd1E/aXMR/iO5m3g5/ctO6iSj+aVGx0kK/9y+Snd9h5Bn7f91gubMhA3SQRAZHo+Q6im QcJh4KiVdX

+5ygFhUWZP7h0Yhhv7d7uSLhiaV1TssdZe9tRvgaBK3vIVFnTchBEgrnv4sndst/cJ

vTtzpfrDW7ulfXW7p3477aiSHjn2OJ1zC24/tb15scCiTEy+UfmlVz6le9bKHc1LT/ukbDdvEPh9

XCvDAHSfAABTD6gan1/xSvVdhhwIJAFBQI5K13gNgWPvo0dg5x1z1z9/ocm9A/vg1eEjGO1bRaPC

bbTvgrzMg4GOYysfTg4xMHCJ0TVK0r87F3C/Jf9g4bjk0qPLKjuXgIs/bj7jlid27IPRAwu4x+gs

yZ59Y3Dbj15Dq5fo3+12i4sB3LFMhuAvWj4uQu3g90sAoEbGttaj+LgPteekXw6/ruHx2xZdwUzL

jn96DV49MOIt3dG7VXrZVff8XIEiuAQdZgpDF24DudJYP79fAgBbFjYfqhDta/L/RP92dJBbUhBo

BMHTf3cA9jy68JqKZ0vueWwY7jFlXzAK90o3enEBnnBAcEFeCfVhjucZxW80RRxUBrrfU+X3rBni

o5TaRj7fB4I8DqAlZYncug10HQ/e+Boc3LOwOo3Ohljq3/HAoLPoXAJW+LIQ7DnrL9G2iPMEZnG2

ECX0juJ9a8d4Lc0rRulDqcIxAxBpFhCYAgNDu+w3yn/AgGBRmiv/87t7DPUrT/tlH/ApjAP8fMGZ nyAyQ3DQPIHB

+lX18e53bS/etwYA1/WvHlLCsgAqXMR3/qqQJqJ7AGIDCAxhQfDq4yPw0/+9Dxal

ObLjx0PwyMuj6POlz/VxO898rq/CLOEIAnqDiAFJ7mYHhR317l13lKdsMcEggIvofAnYNwSEuAAA

eJxAaWE4Z5Yn+g+YpXlNxgtVdjwwDF9/cD9W+YzCy76mn2N519cBAKeKJ+vXWPyJDDCm1PZ6968L

gMACEAeEamwg0ji6lIDhXxrJtgMTAPzkr/fBzu83r8VwocndPz8Atz44GAo8ihp0SgILPxjo4fsB FFK

+XedSORbIfe6//elC8EfScJx3ybBA0kCEw8XcNo0Z5I1EwNhCeqBYNrj/pkF45geLIJis3P3z

1+HPvzdkrDlzhR1r8Y5ps0jxPvDTtm0ftyG8PCLk/ZnrAdzX6O80BIBjAVDbYlOQThRO28CaiBNQ uG0PhMqXh

+AXiyCYsFjlf/Fuo3yZOat3dI6zgIc3hGB9P8/4W0OU9/eMAQxotj99Xn3rtzLuTA8l

FwvoIY1DyINi2XZyvEb5USpf3geP9b8OizK+3P3zQ0b5+32AV4oUT6+CyalJN3T/jq+LUezFEb7h

R1fNkb7x/t64ALAsYCLJPgGTlaTRGOi18g/cvB9+9v8WA8NGcusDB+HPnPJR6SX2JhB8GYRGS3fU

H6wf4ssh6B1BdpygVtvGs34rE9Lt1y59+RGjwM56Hy5W/3GWoQJbxFTSxginXbgE/t0froDWZZOH 1htRho9o

+MoPhuG7Tx52ChRMuUD+PYz1M+dtnz4c82eVDQgUCgDxTWLVZ9a+Z/XR/vaEJvsxiHM9 Hfl8wlYaKb

+eBGCYjadMPPA3/+lXpmi08AaXFGXnnjH4+Ndfc8oH9hYw7cfpox+PRR1A/+6uE+kb

wRQNDLXX5+rKifz9CZvgLZe+uFWAdG+e1OzDOuEBUSdOiMcUpowKswqr/nN/Zzmc/zsL87Uzf/uz Q3DrPx

+E4VHllVtCZQfLRmVj8Ac42tf18nFAUSE+APY5U/Xb9tza0yc0DdmEAfD1nl3taqTVuALd

UY/6AQgMUEP/CiIYFAaKCo9bUKxYmcHHvtgObzphYcw+ttOUy7/y/WF42FT3qFoXXuaQ6fQtX6RU

eh1cWQflx1QvviDKsEZ1tLV1TXXN6gk1zU7KCd/84Re7hJL31TvHfT/f58oP5WOA0HgUts3ym1e0

wTmXL3nDAsH6+jt+aqz+xwdRYRBonStTIcU75WMuD5jzezZQwU0ofCWMwokdhM7X7Fx3xsBEv9Ok

o7CbL32513xoS/H4UYM/BgC7bdsLEwAIPwPZcYYNPmhcQueFM/NCirmQA0bxt//kENxulE90T2/1

FEz5LnIv8TeAqvAyqJgRRJegw+ti0fpB9z277j29k/luUwrDb7nslfuMM++ifR781QcArgsA0MJb

fk7g0CJcY4HQde0SOPsYBsKBw8bif3LYKP51GB7RidVrtNwQ6Lnj0bKT9/6VCm4hgCa089vtyrPn

v7sbJilTAoCNB3ITD5g/22H3ucJpPw0A0RUISCjfK7u4na7fdEIGq88swb+/tg2OX3lspI1W8bcb

xd9hrH54RPkCmUwtFtDfA3vNK0/14ksf6Rj4TqD0bkAGAJMNVE05uLt6lJy/nkz5if71pS91ShA2 Hmiv5/9ron/g1h

+DwTzEAiLEBxqIGaKbsMdOMUD4zQvL8P7fmn/vKrRKH6iOOYsfeGEEW0rROmnm

LhEDPVKk66jBLDv4fBmtnF4FqwuBIr4efmg0V2uq3ZNXvpVpmdQtl/9rj86zO2Hc6L8e/YuQChL9

5yxeCG4BsDdy4VzbcgHveG8J3ntOCU49M4M3zxEzWKX/+KlReNQo/kdPH4EDI6hk+8tsOz2br4+/ uRPwvX7xBc

+aNdzwwBCLPNKDQEkVe/cioJSCK5/rPq0fpijTfnI3X/bKJvN7b0qtv+D/BbdsqIn+

G9F/eh27h/T1BALSiadIB4R3nZHBb6yW8Ja3zgwgdg8qeHZ3Do/tGoNHqqNum4ZOeYv3dG+/kwiK

977dnc8i9Yf3+2UcBAX6LyFwWLu+wgkaHDNA3vfsBZML+orSlCdlMwOjmC2Ngr/IAFyp9ZUcGAJS

oBAbKB1dSk6fww4pueu8qmHJCgknrRZwUoeEtxow2GWpKTu/9S1+SNMJb67/s/cM+l9gFb37NWUs

XMFzr5rtoRyeNesDR5QbOq+wZRSQ5uNahylYo8WjK6C+eQJCwJcoPwSEaOWszKtDyx4E12H+9T3T

fVovTFOaZir/x4AAAgiiMmPhp0HwR8dZgYjTP8UCOrmOfRYVT51VHSikb0zJBW5LiK9WxzW9U9eX

TzV1mgzTrcYpV8FTuvTTaofBseTPwb+RS+EUurVv7tTR8iUERRcbbwgU8Z2/CAh2DdG/+Zl9T3ef

2gtNkKZVXD5114m9BtN9NcUgXUgT60FO0PWiJoVMMgyBFxeancNNNMSXXejCPeIHwraGtF1DCHZ9

AAKCA8i6Ix1rRv9k9cAtX8ZUz/0hl/fH4E+LmBGEOf7CJM8QQYSMo/3ppikfoIkAsPJpAwLzJfv8

nkhWFBcUFVMvZQx9DaEYWCIT6LjPp69351HJ7i2obmCr8ANbNTZlad+o5a7S/PvpGuUnNC85xetE

6bqg9Bjsect295chZfPBW3APEWThFfACG3oYC7l3/lnlX9g85QM0GQBWPn3Xql7zAzbXBoW4iDoK

L16j4zko3MeLSKyfX0coEFD4o7gfrmTHYjsnvhqXTY6hRdHf6+AedPKePYgRv4CQ8kVrhsgWeC0U

4oTIChDYIABRyeue/u3mKh9gBgBg5TPf/fWt5tZrjEerjmfVqVK9BJZmVk7HoNHn3CttGOREZILg

gkjhVJEqIrKOpMqPa82DQOqAATrx8fz9PNE1QOI+wv1ZWhep358zMcmQeXprnr7ID+dutswIAKx8

9q6VA7nS3ea5VO2+1sxXh1Xqz3klMfHzCVtE60/AIYAceVC2SJSegkIERDKkAQRXwL8qFCwfEqqH

4OcxOo8+XqbXKnILgqeIGt1LbPun36OUelSp0ponL3rXhBt3JiszBgArm7+3qqpaWteYH7wtUXTo

Sg41bgDqHNMiBQf5+gAO6zKUp26NihbM8gWjDR0rUsDyUQg3B4Rb0atwBiAwBJpmQWEJouULiC7B

pXDxGBRcRMIYHszbDo+Uu55mQ7lnQgTMkvzFpXs3gBQ3mTStnfcFSFoHwY8tyPEYTxFDLQCKqSO1

M8S0MKSZ0qeCZGm1aSDV3yHk5UnrmlOaitF4gaJJgRQkaor2uYUTtYfePeiiWO6fNA2b0m4uVd9T

l546velDJygzygBc/stdK7fnpbE15tdX7H4M3HjAV2xPwO1iPICuI7leQ3Kf5ANk/WHUimUMoPdk

u1SCsoUQ/Hv6iPfkJ4L1R+WHVE9waocAFMV8fkj9eNrnWacylsk1s6V8gFlkAC7/4/K9mxRk1xsd dBy19IvFoMjYIrV

+AYEZeJtCHnQtgm91zMKLK/galdgHT/t6fLBMFQsykgVp5MezaM3Aonxq6k1T Q84cOrQHKA

+EITt444mPvGPWFE8yawzA5b/9/cqtKpPd5vHuKBgp8GKQ9++15ygmCLEABntaU9Qf c35/QoT35zojRHSF

+oAS7JjAAJEViRgzhPySp3/hz+hYL2CFJB8v6MTyYwAJ2w0WVs+F8gHmiAG4

9PYMdo7kcKd5SB00z0Bi7aI2HtCF63SxPUHEbWtpVB4OpdnQCodrbHQB3vlC8t44zIJDJwx/zk27

lmlW849AUDKtCPJavlF8xRSq+gZwvr65kjkHAMmfXj64wbiALUZxHSn9p+0GUbn1+htgwKh9MKnQ

+oIrEEz5CIAcI3enHPZCpaBo1hsHeOdLSUCh8i7UVgOz1A2g/68YwMy54knmDQBI/nPP4Aats/WG

drt4M7GGtNmY9zZWQkDS/8DGAjqOVHYKd8cREOizFbfqMJuGBt4z1zMAtdxh50zJY4aYMShWFwg5

fwCCCX5F3jdw7fxQPMm8AwDJpp7hrjFQBgxyvUsTecpXYIMQBEKaWvp5DHx7AW+mVeiT7T0pwAMO

gpIOKaLvpKmCsott91DsqZMwAAzZvnpKjG0b+P35pXiSeQsAkj/qGewAKHcZFVxvlNoZ6F7HYJD3

GdA0fQ1A0okEQi4P2BgD2AQMiSUnXa4wK6DeOmGollQhg4jHNO/gWdFCfXusNLp94LrV83rqtHkP

AC4beg51SMh7tMiuMI+5S0Gx02lchxFIeIxydrdNRRrBAjcq2LCAj8bl0bv3wjGp0l673l1UDADu N/HE9oc

+vaoKx4gcUwDgYsFgXEQnCNljAHCWUXQnVQI1Y4OYNkJoyElb7IRP21jFjmICkLGnLoTB

GIEVqrqsKybDuL/lyHB/pXd+W3ojOWYBUJSeHt3eVjrcad2EYYgLjItoN6qyLqPdnk+AQBE55vGe

FXh6F0FgrH3IDrcy2wO6ZfRRQ/1VY+WVSu/xb4hZMf8NIdkRQKhwiigAAAAASUVORK5CYII= ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/png Content-

Transfer-Encoding: base64 Content-Location: chrome-

extension://dbjibobgilijgolhjdcbdebjhejelffo/assets/icon\_white.png

iVBORw0KGgoAAAANSUhEUgAAAIAAAACACAMAAAD04JH5AAADAFBMVEX///8AAAD/3tT918371cpF

Mi3+2tAaHB4YGRwcHiD/4tg8Kyf1z8RGNjJHMy5FNC87LCo6KSUeICIgIiRBLipJNS/30scnKCs/

LSgiJCdMNzLjua7ftakYFxk8Ly5DMCv5zcJRPDYqKy3+/v5OOTT60cYlJijyzMFIODX////nv7Tw

yb4sHhvuwrUdGx0jIiTsxLk3JiM1KShBMzFdSEJUPzn///9YQz3rv7Ltx7vzxbnlvLFJOzj0yr5L

PTrmtqktIyLZrqJFKSTdqZpbUE9OQT49KST19vbdsaXXs6rpuq3wx7z////7+/tQREI+JSHusafq

wrfu7u/Nj34xLS4pFhP///9SR0ZdVVXVr6b2v7bFyczWqpz////NnpEiHh7///9JLCgSCglbNC0V

Fhc0IyAwIB12cXGlqKtuZmZ8Rz4uMDOprK/////SlobLz9KWl5l/fH7bpJTksaPOqaCQkZN6eHrz

uq/6x71WS0pbOzTXoZBRLilDNzfGmo65vcFLMi3grp9GPT3euK4gExEZERDTnIvKhnadXFBSNzKu

sbRISk09MzMWDQygZFfDe2qoem/////EkIEAAADEinujpae/xMddWl3LlIX///////90T0ff4uSe

oKKGhIfl6OmxtrnuqZ7///9eQDpKUVXCoZv////oo5ctGhbYz8+8gXKMio1oR0C3uLppYWGydGZp

PDXChHWrg3noqZ64eWrU19mUWk4gGhrdjX2HY1xtcXZ8aGWddGnZ3d9yW1fRpJZhYGX///98g4iV

dm5BQkbFwcL///+4i4CqjIZWMiw6PD41Nzrnm41kaW/AsLDVu7bnloXKf27lz8pzQTqTbWRwamsX

GRshJCdjWlqul5aubV+HTkXZm4waHR7Nalu7k4uWj46GcW3dyMQaHB9yVVD////bg3IbHyGBWlAW

FxoZGx3Vc2Lf4OCIkZciJSjn2NUkJinv4+EaHSDBdWWThIOQencZGx2ZnJ8bICCYoKUqKjAVFxki

JigyODpQUGAkKCiVj78+AAABAHRSTlP/AP//////////////////////////////////////////

/////////9////////////////8g////////////////////////////QP///////////5D///// //+////v////////////////EP//////////////////////////////////////////////n/8A

//////9gr//////////P////gP//////////////////////////////////cP////9Q//////// ////////////mdf//////yP//////7//MP9I/0Df////lP+0/

+////9g/xDfKIBwgBBAeffJswAA

F7BJREFUeJy1mwlYU2fWx2NlSUoMISwhJCQSIMgaEooCESGsAQQaRIEBWVQQCwooUlEBFXBBrMsj

iHXDraKIdUfrgtW6a6uttnaZz67TTjttZ/q14zcz33bOe28Q4d4EnJmDYi0+9/+7Z3vf894bzqjn

sfHX3p62cbqvL4eY757pr097e/I3z3Upzoi1t07bU8hhtMLp066N/7cCzJ08zZdZ+6nt2Tj53wUw

eSPLnQ/xxMZr/3qAb94YpjplvhuHG4vhAUyePhJ1yqYPLxTDAXgeeeKGrf8SAFZ5z4zt2y/8KSMx

9Z9CsAQwnlHec1dP9+zs+Il1dXXxcUcq2zM82RBet5QL5gHmvsF00cSemtoQJydXoStl8rrs0gxW

L7wx9/kBJjNV/a4dcSFyoZBWh/+wgy95XA+rE3zNZqMZgLnTmO6+rVaOqkJXuRNiEH201oWJrE6Y

9lwA4xlu3/NqdogcdOUh8IecIkEGOzsHB+3sjNydD6tamILhy54JrABbC4deJ7W7DoRDxuF3J2Im

BKFQqynOfphVV5fVvYuBoPDtkQIwuf9Pc8aFhNStmBgCGahQWFsrrBVOCidXV3AAQZBv2ZKVFXfp

yXamMLwxIoC5rzO4v+JR2sRJtWkhqC1wRhMIBIABDBACrVCuhejI5a3xDxnD8DpzNTACjN/DcIGr

j2qzsuMmwX0LBM5ulBEEhUJBCLStWiGUJjCsuNTAkJHMicAEwJx+cx49elg7xtreGdW9iAEBuAGc

AD5w0Gi1WiGmBeSFfEXVhaFlyUjAAMCkz6momTN7Tpo96PfLIwGJAw2g0Tq4gjcAAVKidUv70BbN

RDAUgFH/1M6qqiMT7e3R+V5eHsQoH2AYrOG2HTTFxQ5OkB4Ex04rzyofFsEQAEb9XZXdC4+Ms8fc

69c3UQABalIA1gJTRICgZThRGAwwl0k/o7Sy0nT/brR+KNiECaGhHkiAkgBgp7B2puoCO5Mwi6E9

7xlcC4MBGOqPk9rzZMfD+DECvH/aAShOLNQD9ekksFPQHsC/gg+yTw292OvmARhXv1M72qqyiD4G

AAFA/iUwIo8eh2aU4KBxcLVGoxoDlkXrWwwNYZo5gLeZ9DNKSyvjxtk/dYBJfoKHl0keDWpAYHIA

NgaNRriinGGN3MoOML6QMQBtbXMm2QtIBmIKmPQ93KgCtEsgv+zsnOCHJCFMAFr5JYaFoXA8KwDj

pn97aflOcIA13QARgNJ3ptLdIYEYAChCaSwnJxpAmMbkAt+5LABMCxAnsbO07Ug8pY85QOuHegmo

NYDWB3O19phAfmIiA4C6GqZdwjRmgMlM+pxTpeXdtZABFAEAUPpu0Hwg8cDSwWKKlt5cvHjNmjUY

n1D0DaYFtOa6LRVMF53MBMDYATipneWlD+MJAEkCj1Ba3xXvEW4/PWbmK6+9uanPYPRpPLxmGcbI

C3xAuqEQdosM3eiZIDwFYKxAzgVwQPakMfbWVA54YQAmhLqR9Q/U05d+cPpkvcHIH23zwgsv+DxY

s2zZMrJQow8cHOQrtjxh3Km9MRRgPKO+Z0Np6ezaif0AGABIP3J1B/D8K68dypeFu4D6i2Av2NTv

XQalKHCmFyhh3aVKpg0ShzN+CABTC4RFoLy08kgaAGAZCogDQJ9K/vSYoygfDHf/ImXghPDzy6zJ

bonKQ3n2E4ZuCDZ9MMBWxn/GaS8v3RkHKQAAtANQn+R++s1P39yUp/OH23/xxX4Cm9H1i4sU1KqM

S9KK7lzm/frkQQDMc39iS/mO2VmTTABemOPgf9CPKTp6+lC+wc/H6hl9JJAdLlKQ/ohrtPASw75g

oAs4Zh1wqqW8+1GaCQAcQPLPQZNQdPMVk/uf6iOAjc1oK+N3RZQ+cUEpy7ww+RkAZgekNrS0zY6L

hyKwx72AW+gErD8HDZTep5S+1SB9AuDicz6GLA9YKsIalqlt+kAA5h7EyehsWXikNn7imDFkKfAK DfWi9G+iPlT

+YH0b0AcAl5QNRWR1sHMVarMuMF+cdgEFwDKBn+os3ZkNKTCGjkCohwA2HpqYm6+s

Ptmhh/xjCMBoKwDwCd5ALQ8wLrR2sgBMfwrA3AM4qe0tTzAC49ADWAOhblAAGmh9qyH/pcHogBcG

O8AKAPh8f+P5GEIAc9xOFgDON/0AG1ki0NDSjRHoB/CyJvofnP5kc5/M6O8yEOCFAQ7g8338dWcB

wAHmZuEWthOMN/oBWM7eLjSUVmVnIQD2IejDzk6QAEVHIQE60AEEgEZ4qo8O8PHxD5dtw7UKxpR4

xlENrNAEcI35554VPU8e1qaRCEArhs2ANew802e+cvrNzX0GXbiPC1kACMFAfRfQBwBj3xqNBnqh

vC6XLQaTaQCWCCS2t3Q/yop/CiAAB5QsPfrKayc7AAA9MNqGQhjkf9BPCTfqjqcDgVDeWsoGsJEG KGT

+8a6G8qrsNBMA5ACUYEn60qUzjx7e9t3ZDdf7dPwBBP36Pv7+IB8ebPSTfUcBLGQDKJxLAFia

AKTAjodxNADZDDglFDc1NTdNDShOcFq2Zu/hbedl/QS0PsiHBwcbiflJ85dhDFqr2AAwBhzWCHhW

dFYegRSYSAMIwAFdM5q60pcuPgy2GBG+1gOADZYf6YCgb8y7fn7Dhuv1Bp2fn1R2Flwgb61hPT6a

RgCYZnGw1NyWnY9q4yeRIoAUECjsNOlLoQRObsrPMxjyNj3YtnjveXQBrY/3Lz1/eC9uzdYs/u64

TNqILhCGzGY/v0IAli4EOVheQ4rQBOBklwAZePqTY/V5BqkfuFiWf77ehSag/X/98OI1i/duO7wX

xtbDj2U5Zx3MAsC+hMNWhJyM9h2z4ygAEgKIQMzNDz597c3N+XkyAAgOh3BDM7IajUaVf/jx785f z

+/L68vfdPzsXo8rOfUJAMAeAhhSOMybcQ4WwcI5NADZkEEXLIIeQAAMUp2fMdzfx4fvQhNQ+R/u

J5NJdWhSmaFvw7YH+r0OwtYaVn1IAg7bQsTZ3gN92AQAU6fCzgEyYPXpTw4RFwBBylMCugDCjX5+

Op2fDr/rpIb84/oHGruQbnaAPQBQyPKzC5iDWAQAgOcOkAIQgdWLnYo1GvuXzuZL/SAICAAEVABS

sPj9dJQHwBGNhpycxwl2dU/YAQpHcb5h+9mp8irsg1iFIeABkgI3i5uao3szm5p7A6K/1kMe+FMu IA4IN

+lLpTIZhkLamJRTvcxuRQM7AGc8h60NeZ5qwyKg2gBuc101TTNmrFunWru2ecaM5qamAPt6

HUaBTwNA9zf6YeyPn/3667PH8/Kl0sbGpOq9drXMG3PKrnEYJ3IEqNgxO5usBCH29rDRhi4wc3Fx

8+27b7nPmNGrUkbfzrbv0wWnIACmgI8/3L9UarjiFB0Q0Bsd7fWAAOTsdapiLwIOyLMVgWcFbEay

iAPG4E7fKUFz9NPvVFsu34L7L1ApM7U/Vk0wGCkA3AdCAkplfaEBvc3reu/eXSHXnq3GGHgIWBdD

tGkc5oEEASofPgPgEPPB6tfeeety05aa3gK1Sqvdklt1PCwYALAKIQWMOln9mIDo6KZ165p+/PF2

tHbclb6kJOcvzDmAs5HD+kAmt3JOHLUSjLF3QoCiD17bkF1T0/Skobe5t6C4eEVFz7mwFFyUwVz8 g/2k

+nNr5ZkFkCLr3mn4MVP+9WPwgOJj9mN8sOkclpUAABb2A1AegEHw2LmqmqaAaFBoTi+qrcj9 DR1ghZ0Y9qF

+0sc76tIq166DGGkbOtdeCTZCEiqu6/9ixge+HLZHoQMBsBM6JSyFWeR41cKpU2cA wNSZRTUVufvJ/cOCTAHcKE

+Yk7gQAJq0ne1z9vsDwOO9fTLZ++YAWG0IwEzYCk25UVnbNbWpuevm

GterFQeppQg2ZTaYhLr9LelF2VoAKBnX0H4jHAFuPNBL/fx+NpsHLFYxAEBBAax+89jms78qStJj li5N

+PzgQbInJDtjG+zExvdy3WK61q1b17wsrqKh2sXfrzHpOGzewoxG9mdarEaqoB9AIaAB7n8y

b3PMzJj0lz7+km9l2o69+OJoshWMfP9cekxX09SYNTW57/rzwwEgTy8LCwwPNhMEdg88LUPwgAlg

ymd37tzZvXv3q79bNHYAgJUPH3qyt897oTFFRUuL0nre1UWmGHUAYJAag1NS/J4jBqdMAPb2Tz2w

at5nBw6c2P3yywSA3pbDfObij2uC99hF+vNHY9IvfpyzIT8QGlNjkl4G3dLfX2fm8SqbXaD3I7gn

Vlg7K3AiO7nqs/knTrwM+gjgTQHYwB/8YB9vK28A+N2rL+/efWDVgQOxybAcJhkM0rCUSB9zAKx1

sJ2sBZQHrAnAB6tPTvks9gSR/92iRd60B6AN2fiERQ4EAC8tSIZFMckQFRboDwBGthCw9wHO9rbZ

MBUMALiJAPNPLKD1wQFUFVgBSaQuxdvKigIAghPgpijQT1oQFRYc6eMTzpqEe1hbMWdXWw01F1Gn

E9YxeCI3Zf6CKJCPQH3aAVaREWHJydJgFwIQgQRgC6KiopKSkmTJgRgBP9YynM5hmQpgU0ptSExz

kXXMzKOffgIAyYEREYvGov5oMhLwk5fHxsYaYEyiY0ARREWBAwxRycaUSH8/9ip8nXU55mS0DARw

E8QsBoDNSxAgEvS9yWYU+qCLf1hyVFRyMAUQCS54NRn1o2Q5OcujdIGBYbL32YtwGuuGhJNIzSUm

AOeio7AnP7RyeVRYRErkWL63i7dpJHDhR5Ktmbe3Nz8yMiLw1eRkop/UF7vcYNA3/oVVHjckbGMB J7Fz55H

+4RwB9m7DqSgWwhrhj33HyjQYwlTi7Q1/dfHmA0BKRCC6RGZIqs7Pr2780txaCNMhh20w

4qT27KQOqDALnb3cXDXbzsJQ0KeXhgWnUASUvLf3okURY6mNaaR/Sgqlr8/5+Ou/JKZa6oDfsG/L

UxtoADwkdPbyUgQIHmzYQM1luBsGgpTksZGLoPAPHLizO8KbjCa4NQ+Twv3rq0OzLfdf2Jazzaac

1PbuOXH9AB5eipKSwx8/vt7Rp5eRqQgI+C+fuf8fYD/88MMf7h1YEBseTkYDGEkM+qSLJZ9b1OdM

NzOaebYvRADSBwRuHh6K6JLic+/ur8+nXYAEPhGv7j5w78wffgCK+2fCwsNpfb0+51xJcY9lABzN

WE5pcUv0MI4+InN2C0WAEs1vV6vzwQU6IzUU8VNO3Lt35869e/fO3DsQFoxmhNFAr0/6oqukmPmc

/Bm7ZmY8J1si+ogMAEIVxUDg8OuXSdUUAZnK+GF3zvyB2JkoYzitb8gB/a6SkGHsQnA8ZzukowCw

FQvs8ZjYWgMAJZpfD76Xk2cAAmom4qcsmH/vzJkz88OCIQB4LoP6F7umdhXXWs7BPeSEhC0JyJ4M

PYDHtKETBBoYeQICSmYf/DJJb2jUBfvTQxk5l/KB3zCawECqz9nv0TUVAOIsA2w0e0hFAYyhQuAx

wdmOAAQExJeCE0wE9OEEUpDRSJ+T81sJ6E/t0jyyDEAdUs0tZP7pqSe4KTQd077k5hpNExTPufol

bnWMAwj4fP9g6vYnBHRRAEcs6heaP6i8UF6VnTUJZnNr7AMTvJy00cQgE+zOvZeUI5P6hftQxxMu

MJsapYa8nP0XSwJMAHMsApgOKllikIgEafjOkMJJIQj1EraCyeXk/Sn5ho7qHL2BLgZIgmCdTJ+3

/wtX4iIk6NI8tAhwzXRYXcj2LxJ35e6oyV4RInRVOCuElZWVn3dXPXwUV5uVtmH9+s0rYc+Nx1V4 LiDTV

+//QhHQb11Tiy16wLf/tJz5mSVlnqmJ29vbarLHOQipgwZP+D+JGfWH1q9ff38KMIAZ9Hl5

j8+NKQb3l5RAkDRaTUlXicUq2NgPwHpMM4Aj4+C7A8fc+kOrVt2/DxCrpsybN69j0+NfZz9KWyHX

arWZmUKIkzYgYJLZsZhDP7w0+8jGDM+mY5unTAGG9f9J7GQ18VXGnyqu9pS3tZW3tPRcPWjBAwMe

2bC2AlZL3QT686YAA3ph/fpD1SO9wrMPrUbsggwAmLcSnI8I9+8fO7R/xPq+zzw3HKkLft4E+kuW

LPmMIKw6duzxiIe/rc8+uh2hC97b1LFyyfzY2PlLlqxEhs3XRzr8+Q56djwyF6RWT5kH+suXL0eG

lUBwbKQnAFsHAYzMBT+DA0B/ARiF0LH5y5Hp97/PZOEFBmZLrd5E9KPQEAEI6kcGMPQFBrPtcJC9

v2ke6MM4hEYQ5q+cMiIXMLzCwfISC5P93DFvSewCkA8jlkwRdIzgJIrxJZZh5mHi1fc6OuaD+5PD AgMDI

+A3zMaIsLLjxufbh8nA+BoP+95soGVUPeiYB/rJqB+RkhIBkxiFELuk481xl9tydyVapGB5

kWkYQUhsX1u8eWUsuf2IiEjKIhABArF8/rzTsBSWrb27s629Yjv7C+++o1gAmF9me2qp7W/dKgjY

HLuAlh9LGc6jxA0LYjs0Bb29ve69auXtd/bdrWSOiJmX2ViHFCKfe/mWOiio9wE5ogB577HelCED

+iEsasGaXnd391mzZsH3IJHoVifThd4exQ7AngapFZffEYuC4Ppey8PwhIIcUdDmTfshMPkwAnDd

kYLL5dreZiAY9Ir1cF7phNjnXl6rCuLxuO7NzQVXAiNNBySj6WeGeDSBDIETet2DAAAR0Avc20PG

MwuvdI6ayzAse+bW7FMGBRHnururox9HmuTJSyvkrQErEokweYF7EJo71xY94C66NGhb5Gvppdah

r/V6Vry1r0xiC9qzZnHh/gpUExu9+x8Z21AMhMB7bH50QZCIF8SzteXxeLYi/PXkmUS0/FrvEILE

7ndui1Gfi7fEg8urlC8ZnwHoJ/C5oioQ8XhB5J/akl/cWxVm9S2+2p14t0wlEdlCRPF6XABQFygz s4wUQP

+rA6PxmIjfGKIuEIkgU2h1cAFXtDbRnL6ll9s9W8pEeBnIK/Qqz1EtEivVysxJ1SQBbEwv

L5HpLPhiQZBaLRKJIAyOEAYMBBBUmtNneb2/nyDjroTcB9wPyIscJRKeWiwuUGVmnst5GgZSB3xZ

nKqggOcokjja8vCLh3GwnWVbRj+43DPs1/sHfMChZx/Io+sJABCIeCqxskwszsyUx91o5KMf0P38

lKQrmSo1z9FRDQQgb+tOKACAy3uLbNdG8gGHUf27gx/3KXkkmiLwJxeuL3FUicVKlVIlBi+0xl/8

7QbY8SsXJ2rF6oICtVqtEonQ/TwqCxGeK8ZHl2yftGH9kMvbhZiCl8pu29rifYM7RRKJSqJyBP0y

pRK9oFRm4heaWKVWF6jhmxo84EikeaQT8PDbvsRC1s98mf+Yz/Z94jL0O2aTyNYRHKBUQQiU8KsM

5ZUgCSCtSrFYrZaoVOAfgOXZYjvmwpIwiwshcOeK/vYcH/MZhQtD+ztlIomI9BVoARJHzEIJ3rqy

DL2AphLf6uysgdIQqUWOjlCmkCmo6s7lziKVC/3ww/9iFzH/Ua+/3SqD2xdhJVIEoO+I+uAAsRpv

W6VW7dxxufSW2BH9I4J/Y0syD6RhPYJvXO73fzSnYf7Dbv/7P0osQx4pRFIGEokIXI1hgHwQY8RV

d38s77ytVAGeGgAgA6AEsHFADNAD/zBz+xYBRo36+59JAUIfIu0N/1uiUoqhFMQS/C0RicWXLm9R

KiU8cI7EEXsAZp4tF3s3j/fV7y0IWP7A41+/p5ShpfBmQXUDgARiIAYICQCo4d4BB/o1qKswWu7u

thh48AHvq28tXn44H/n86/dYVnBLJBUxBCAvVmJRgiCA4JcYmwQwkBCQ7HMfhvxwP/T6x69IWpOO

jL1Whe0IUwEKA7oDdCaVytYR+wWuRZiEXO5XZnNvhACjRv3+w+8RgKyHoAiC4HOMOvoDHQIAEmwC

Irh991m//MNS7EcKAPbth78AAdQEZiCISgiACksTKMSAhg0bKu+XD4d38yMFIAwfYUvAGwXn87Dx

iMgaCU0AyhV6Bvf7D/9ovu7+KQCw33/74Ve/qHBZIEsT1L2Iqkdbx18++umvw/X88wMQ+/v//fdP

f/7zRx9BnwYPfPTRVz/99N/fjlib2P8D/Bt+Qh1LbB0AAAAASUVORK5CYII= ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/webp Content-Transfer-Encoding:

base64 Content-Location:

https://gips2.baidu.com/it/u=2096770753,3653284586&fm=3028&app=3028&f=PNG&fmt=auto&q=100&size=f48\_48

UklGRoYCAABXRUJQVlA4WAoAAAAQAAAALwAALwAAQUxQSOcAAAABgFXbdtzmQhCEB6EQBMEQBM

EM IgYJA4dBykAQCuFBEITzoVZTp9+ImAD9ohYX2oRwqyz1mw2Ys9yty5wN3XouoOa08OFA6TDATUvN

AfvqBJIWR+D8KgNaHYC8lX6pt9QfN8j0lz9AsMENlv6RUpm5JjPzZ0ll5pql/6kjbGbkzS7qZg5R

0tN3SUBRm9IeBcCa3gXhLLR+2ViYEe+VzpJ6TiBOOBh97zDAbUzJu+rDOlQAv6fuQ2181s/KEdRt

zvitkcybQ8PmY7yORhn40ETLPgR+NylUSDMkWZxokl4QJk2PXNo8Wx8AVlA4IHgBAAAwCgCdASow

ADAAPiEQh0IhoQnOApIMAQJQBb9dDYt/ifW+AH2zbqBugP1N/UD36uiA/gHAAeWB/gPd9/XT9Vfg

A/Rspe/itZUmFSBIIRipY+31co975UVbOAAA/v9NjCGWldQAHKr0TwsMJUvVz0mbjQpPX3vf2KB9

nAMxbsjpUQHsGRxaw5kN8TSV+0NdWYrcf/iJTkNOum2GjGFiCGV6wgr6LWjnJcwr8iqd/pY8t21P

VHMixgPKmfQ6cAcLTczUmy4CV09c/Rmr3upM7rARcCG5FBPwKqWe7HLnOLVKrb7OC2gi8E7/+9n3

ftLXZbm67rga+RHdS7vjFJyzQo0r8YCnagt565qaPE+fWkV5h1N5paFq81ciNjz7lXfHb4gEP1iQ

5mvZCAtGE2JkSloEjP/t1xSkUJYadIW83rf/qBH/r6rutp0PjoVkdiCf50D3zEjXlYxR6rR/3vfH bVcZ

+dtXR1Ts7IFaBzFLnlZgedpsMqBd6gAA ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/webp Content-Transfer-Encoding:

base64 Content-Location:

https://gips3.baidu.com/it/u=414863231,1416682463&fm=3028&app=3028&f=PNG&fmt=auto&q=100&size=f96\_96

UklGRqQMAABXRUJQVlA4WAoAAAAQAAAAXwAAXwAAQUxQSK4FAAAB8Fbb1rY12rbtEg4JSIgEJJQD

DgfgJHGQywGnAyTEwbU7KByM/mP72A9ICYiICTi+6e359ef7n/Nf+Dw/3n++Pt+O/8zb/fdpmv0/ v

+9P/wG3tw8I0dBi/Lg//V3PH9lMyGbG93/+mtvbadkwRousz/vf8fqvL96I8+X7PZ8iNTQJzWoS

hZz37/X0bpqHR0KG8ffTN3r5jIwRSrMmhnUU5/273H6ikO2EqhkNTVD9vH2LpzNIxggxuZrImDif

vsGPzxYo86xDQpPMQ84fX/ZiO2OhEJF5lhVEefmiV01qQJMoymZIWYaSty95hRSaLCvy8ITMX7/g

xfYlKNpIyDise3nYj2ZNpi2mmYY0y2az+PGgp5OG7dqahyQR0mT38+kht1MuZxY1yTokqEUI5+0R

P2WeRMaGsRDNMp0orYifD7jbGDc2o2G5QNnP+HLp6dxJWRcS1F6abDYh+ny68ps8suxHNBBSi1x8

v3CH0JVHhhBkdxZNPO+dGYuG0iTSRpCxUBsEDeHcukPmDdZlbDZdGEwbMi1RXndOiSb7QeYtyJj1

EFSWfd5Wd2QeLcqjs2wWMs08vK7OZmVcbbeR65lGpjXpXPxjXsi8ScjYtUSWMUvI+Dx7HyplbEAQ

UQhJDWOoQRmLzPuYPCEkZUyW2c/FyMI6m7fhbowgGiJkjDYamoxBmRaFkLfhI9nPNEXDA9NGMAky

zj6O43iCEE1EttMl1CSTh9+O444ESTJGK1KkVbkeaqPhfhy/ydfneg1dCFoofh/Hn1BSSEhoQ5Pa GAtCcjXjedyQ

+YCCZD/ZDtJMpBBlfXvOvCwzv+LKWMbsh0nh+Q0ahEKUR9ciaKfF1befgyKClIfH

LMn1WjXi53vIPELQokWzJmXs0naQ/31olulwcSUhEw3aaivLP6cx26k2NmcSSirtlDSh2fkpokko

PMQky5RcjhrG9Gla5gm112rcUNEDMoYgKVKQEu1sZrOQh+6IIMs8Mj1kzMOirKNJpnXBzuVHkc18

uthDvnuWVT5PImpxuZ2GJj0ql88/InhMps1ELqaN0NAQ8fFOxmReq5Dd7DbZzrRZKO8/bScPbTU2

Kctosp0Qfr2iNKSvy26T3agJErw920wu1pDNrJNCO4g0IDzfUJNpW7vNtEIIauNy3Y5TNEkeGprp

ynRSulZ/juP38MXD9YSayLQWNeH3cdwhDT1IWy0aNhNEUWi4H8dNkS/vynaENGxOno7j+LAbXSqJ

9lqFspw0CX0cx3G8acMjZNkw39iMpNX0PtwmraahZvNc7UIo89SQp+H4MEYayrxJZKykGbXKGCGb

/zumzw1yvYhZpqXZboRoEfHP7DjVokgaaCPLoZCQhrGMQTiP5StkmWmkILJDpgW5GjW8rG6fNotk

t7SIzDKP0M5YOo/NN5QGMo2QbAahaijXy/Rl5ziDMiZkns0UWaaGqCEais5j+zkpuxlDi6ybzbPO hMjL3vE

+QIgQMjaoKzWMTagJ+X1cfPo0LTFDRGbGzMsY0RCEOJ+uHPeZGjJmmsqyRWZmU0Thflz/

RcZBJUSZR9YhhKwT8ut44O3MZmWazQqhLJNsZtr/3x5xPH2SZhCpmiFUtOFCovPpeOwPmWQemRZJ xoKsQxPEj

+PRdyWPzNXJPPMhQ/fj8W+hnRZCqKEhBE3G2evxlW/GVpvDdEIp00yD8HZ87X3QKpHL

RYjUZNn9+OofZ9GQEDQkTUyDEoX498fx9U+nIdNaRcaUCA2EcD4d3/H2S6IJzQRJUIVs/7od3/R+

kjEhZJqxkGVNPl+O7/v0f9lOFBIyTZP5+9Pxre8nLUIqZGiyfz4f3/5+IjUREgRNQny+HX/l/cw6 JVTmzXK

+3o6/9Z93JFmHGhB9PB9/9dP9A9mPKHy83Y6//+n++0+TZgTn7/vt+M+8Pb/+ev84/x0+

zz/vv96eb8c3BVZQOCDQBgAAUCIAnQEqYABgAD4hEIZCIaEKdLqkDAECWwGcGaAJ5tGcefqn5Yez 5Vv7N

+Df3L/0mZ8eKfrX+v/qH5X9qzzAP0k/3H9S63XmA/Tv9rewx6AH8v/n3/n9nf1NfQA/dL1V P9F/3v

+P8EH7Q/tp8Av62f+brAOAv6qv0gdPZ7DZw77SfRflu5wdqD+q/krwBOKv8rpHP+k4waYr /UfOdzfPRf+/9wX

+S/0v/idiH0Vf2VNVKrnfwj9hABrEy9YiZIHciWR5CZnj5mCtBKV80fzgjfJ0

979uyOqL6ec/zWtvOL2Avh9H9V1ENeBy7//IAmCnWT028fiM82Me5IkTXaNp2u0tD93MTVA4rGmO

CCX37yh5KvUJp+azRpQMAAD+//V96j/8cQzq+B/QipqgmZY8UOqMz4FCr+/Aux6tQBfpxANJg6Yn

Nf4qH8Yr9+FxecDtstOX2gsauyyeBwAcitmrdBOF52eSfUWlneCByd2oHxSVgEDbB/e408H+n3Gq

6C7YQDkI47uub7dCqyszvHVOwFJKheXc9rlAB+Vb+lakBqhkr4xWCI6njFGj+pHEV794lAFnpHBB

AsSnZRMYhYqeIaRMEPplCwAn//V991kYAjzOj7u+JiopF/GwMmN6pAK/V9OOdituSjZYfaJutdXW

YkfUaeLcypGvwdSNqwpSOa3jrWaa+tgZtrZJFjVx1V+P50z9TooZyOdCBO9kNxIGCkgTGpwTZXR5

/duJX9KXjf8Hk0Rb9talcSLF6ysccSxHyYrxZMOSjeI6w0fEpSziv6XqBASZfS17ijwWyRXDEE8a

n7M0ApoANw8lVCPcRoq125VpLlwSvsw8DO/ri25mEKI++1mvxrjFOkYD21rWgEUrC/+Bt56e1mMd

qiBRMlYFPFoJiY2xkKAcGdDNY9w+lcoAbg8SjYKRSF1FA/7KGoSKc5P4a//SFxf/BLfE24U0cS8t Sol5CM89Bv

+OoMv9uf4GZRuQI7O+XiLRtHf0P3eUfGQMjcMU8/gj5Uy3ydCxQoo2w1coftu+C8gl

JZpeBP99Fq/RrKtyEqm3GNKcXAbdHp02QtMMrJ5KWvzNg/fuQOnBvLc0lxLLnFJbkQhniFh7ZIbg

a96qOMrKGgwnoS3w0cuWlJhZa/gVvxi/HqfyepBhVdtlpHZ6gEN+wDg7/QOenmft/E37+DDyH1nc

sKqfm3IDr7CZYrRwV3NaRVKs1LkUUqzPHXMitIG952NdfFG11DTTAb1CyuilbL4dhVKHm0c7N8UB

BAcU3gbpF7er1gZu9QtH7/9sTvBOAsIDDZm9UeU77HGlAkg/q5nQ1B+9yowvP+qGaQ6mCss6xi08

63kXV9bI/VQD9EWFLIDu7Pmb5WkOGnFaUy9fxg0ct4mBx7k/gBzDe+ZOdGSVahr64HSonz/6fgFh

HISMneznkSAp5ev5VGYy4n8GsvcdRS4ffnze7+IeXoZ7n9uilWqflnxG6/Du01r1MJ+fgtcHo+Fm s58xN

+Xtor3+aBKLBYI68q/W4eGgNHrgHmakY14HsV8L1Cp+3Ly9uHLuE9RVy4g6Z/aQ08KWqbCe

ATVlIwalNAtWDnZjmPFDV//6V+aKrmHB/LcxcNj3N2gIvwrX9/CHj4bgTpsEDdapUGbnDk/4ahWD

vXBY9AM9bdRnl6iASHbPLkbCs4/Cd9+NvOMwbULof7/bYnIYTOoHAHquqHU1gPYmyEycsxN1zJ1i

YjEJiwKlPyd4IGcftLus/DQwijgYnf+vmE208fWTikCHBOP1xp3/j3M5lJPaANie/YqVvdgNP8ka

ChzTY1o8F9/qsVq5g7jiq6RIOvVy23web9ACKAHP7pZ13tgtyowqsEus/qsYz6kUk4vc4l6Cg+Tz

Rosv848kSnJ9IWgzronEUKHS0s/d2G1FdCbhBNuiUCB5m3Em9qnCtdt5H8SyoZjKC+ZzuYYiMm9H

Mi7WToPMxOh4oioDCsjbo1rfd8y94YGzPKX/MQWW/kx8gyXUG98ReXR1rz/u95z4ILbMEeBLCAeL

qqlerEw9xUxgZ/R3BqnBHAIl0J0i3iezRooJjzGj2jOABhNDJUC67rAuICCagwr+3nODzuZxNnaq g8gs+

+6f0CXk9g5FkkxA//nywGR7LdXo6+vXs1M1SXMLGmI5gchT0Pvj5u9AGQA/GKv4oAwx/peD

HV1xduSL1PhTnHYwAd8GUffGSFGqGEF39zYF8Vz3QNEr8aYfURoKDaonuJ66FViMwce+zxGRAC47 EheA96yb

+I3oL4zBHx2Qs5XKju9lyv85IM3H/7GP5W1SYIgGmGMwWpYh3+YU7HIpuAAAAA== ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: image/svg+xml Content-Transfer-

Encoding: quoted-printable Content-Location: http://localhost:7860/assets/mcp-DNm9doVd.svg

=0A ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/ApiRecorder-N1NZAe7o.css @charset "utf-8"; .api-

name.svelte-sy28j6 { font-size: var(--text-sm); font-family: var(--fon= t-mono); color: rgb(253, 123, 0); } .loading-

dot.svelte-sy28j6 { position: relative; left: -9999px; width: 10px= ; height: 10px; border-radius: 5px; background-color:

rgb(253, 123, 0); col= or: rgb(253, 123, 0); box-shadow: 9999px 0px 0px -1px; animation: 2s linear= 0.25s infinite

normal none running svelte-sy28j6-loading-dot; margin-left:= 0.25rem; margin-right: 0.5rem; } .docs-

wrap .sm.secondary { padding-top: 1px; padding-bottom: 1px; } .self-baseline.svelte-sy28j6 { align-self: baseline; }

@keyframes svelte-sy28j6-loading-dot {=20 0% { box-shadow: 9999px 0px 0px -1px; } 50% { box-shadow: 9999px

0px 0px 2px; } 100% { box-shadow: 9999px 0px 0px -1px; } } .api-count.svelte-sy28j6 { font-weight: 700; color:

rgb(253, 123, 0); align= -self: baseline; font-family: var(--font-mono); font-size: var(--text-sm); = } .api-section.svelte-

sy28j6 { margin-left: 4px; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI----

Content-Type: text/css Content-Transfer-Encoding: quoted-printable Content-Location:

http://localhost:7860/assets/Index-Dko6XNbv.css @charset "utf-8"; button.svelte-qbrfs { cursor: pointer; width:

var(--size-full); } .wrap.svelte-qbrfs { display: flex; flex-direction: column; justify-content= : center; align-items: center;

min-height: var(--size-60); color: var(--blo= ck-label-text-color); height: 100%; padding-top: var(--size-3); } .icon-

wrap.svelte-qbrfs { width: 30px; margin-bottom: var(--spacing-lg); } @media (min-width: 768px) { .wrap.svelte-qbrfs

{ font-size: var(--text-lg); } } .wrap.svelte-10cpz3p.svelte-10cpz3p { position: relative; width: var(--size= -full); height:

var(--size-full); } .hide.svelte-10cpz3p.svelte-10cpz3p { display: none; } video.svelte-10cpz3p.svelte-10cpz3p { width:

var(--size-full); height: var(= --size-full); object-fit: contain; } .button-wrap.svelte-10cpz3p.svelte-10cpz3p { position:

absolute; background= -color: var(--block-background-fill); border: 1px solid var(--border-color-= primary); padding:

var(--size-1-5); display: flex; bottom: var(--size-2); l= eft: 50%; transform: translate(-50%); box-shadow: var(--shadow-

drop-lg); bo= rder-radius: var(--radius-xl); line-height: var(--size-3); color: var(--but= ton-secondary-text-color); } .icon-

with-text.svelte-10cpz3p.svelte-10cpz3p { width: var(--size-20); alig= n-items: center; margin: 0 var(--spacing-xl);

display: flex; justify-conten= t: space-evenly; } @media (min-width: 768px) { button.svelte-10cpz3p.svelte-10cpz3p

{ bottom: var(--size-4); } } @media (min-width: 1280px) { button.svelte-10cpz3p.svelte-10cpz3p { bottom: var(--size-

8); } } .icon.svelte-10cpz3p.svelte-10cpz3p { width: 18px; height: 18px; display: f= lex; justify-content: space-between;

align-items: center; } .color-primary.svelte-10cpz3p.svelte-10cpz3p { fill: var(--primary-600); st= roke: var(--primary-

600); color: var(--primary-600); } .flip.svelte-10cpz3p.svelte-10cpz3p { transform: scaleX(-1); } .select-wrap.svelte-

10cpz3p.svelte-10cpz3p { appearance: none; color: var(-= -button-secondary-text-color); width: 95%; font-size:

var(--text-md); posit= ion: absolute; bottom: var(--size-2); background-color: var(--block-backgro= und-fill); box-

shadow: var(--shadow-drop-lg); border-radius: var(--radius-x= l); z-index: var(--layer-top); border: 1px solid

var(--border-color-primary= ); text-align: left; line-height: var(--size-4); white-space: nowrap; text-= overflow: ellipsis;

left: 50%; transform: translate(-50%); max-width: var(-= -size-52); } .select-wrap.svelte-10cpz3p > option.svelte-

10cpz3p { padding-top: 0.25rem;= padding-bottom: 0.25rem; padding-left: 0.5rem; border-bottom: 1px solid va=

r(--border-color-accent); padding-right: var(--size-8); text-overflow: elli= psis; overflow: hidden; } .select-wrap.svelte-

10cpz3p > option.svelte-10cpz3p:hover { background-colo= r: var(--color-accent); } .select-wrap.svelte-10cpz3p >

option.svelte-10cpz3p:last-child { border: no= ne; } .inset-icon.svelte-10cpz3p.svelte-10cpz3p { position: absolute; top:

5px; r= ight: -6.5px; width: var(--size-10); height: var(--size-5); opacity: 0.8; } @media (min-width: 768px)

{ .wrap.svelte-10cpz3p.svelte-10cpz3p { font-size: var(--text-lg); } } .image-frame.svelte-1hdlew6 img { width:

var(--size-full); height: var(--si= ze-full); object-fit: scale-down; } .upload-container.svelte-1hdlew6 { display: flex;

align-items: center; just= ify-content: center; height: 100%; flex-shrink: 1; max-height: 100%; } .reduced-height.svelte-

1hdlew6 { height: calc(100% - var(--size-10)); } .image-container.svelte-1hdlew6 { display: flex; height: 100%; flex-

directi= on: column; justify-content: center; align-items: center; max-height: 100%;= } .selectable.svelte-1hdlew6

{ cursor: crosshair; } .image-frame.svelte-1hdlew6 { object-fit: cover; width: 100%; height: 100%;= } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Example-D1sBsWPM.css @charset

"utf-8"; .container.svelte-a9zvka img { width: 100%; height: 100%; } .container.selected.svelte-a9zvka { border-color:

var(--border-color-accent= ); } .border.table.svelte-a9zvka { border: 2px solid var(--border-color-

primary)= ; } .container.table.svelte-a9zvka { margin: 0px auto; border-radius: var(--rad= ius-lg); overflow: hidden;

width: var(--size-20); height: var(--size-20); o= bject-fit: cover; } .container.gallery.svelte-a9zvka { width: var(--size-20);

max-width: var(--= size-20); object-fit: cover; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-BdvHG3t0.css @charset "utf-8"; .wrap.svelte-

1kajgn1.svelte-1kajgn1 { display: flex; flex-direction: column= ; width: 100%; } .head.svelte-1kajgn1.svelte-1kajgn1

{ margin-bottom: var(--size-2); display= : flex; justify-content: space-between; align-items: flex-start; flex-wrap:= wrap;

width: 100%; } .head.svelte-1kajgn1 > label.svelte-1kajgn1 { flex: 1 1 0%; } .head.svelte-1kajgn1 > .tab-like-

container.svelte-1kajgn1 { margin-left: au= to; order: 1; } .slider\_input\_container.svelte-1kajgn1.svelte-1kajgn1

{ display: flex; alig= n-items: center; gap: var(--size-2); } input[type=3D"range"].svelte-1kajgn1.svelte-1kajgn1

{ appearance: none; wid= th: 100%; cursor: pointer; outline: none; border-radius: var(--radius-xl); = min-width:

var(--size-28); background: transparent; } input[type=3D"range"].svelte-1kajgn1.svelte-1kajgn1::-webkit-slider-

runnabl= e-track { height: var(--size-2); background: var(--neutral-200); border-rad= ius: var(--radius-xl); }

input[type=3D"range"].svelte-1kajgn1.svelte-1kajgn1::-webkit-slider-thumb {= appearance: none; height: var(--size-4);

width: var(--size-4); background-= color: rgb(255, 255, 255); border-radius: 50%; margin-top: -5px; box-shadow= :

rgba(247, 246, 246, 0.737) 0px 0px 0px 1px, rgba(0, 0, 0, 0.1) 1px 1px 4p= x; } input[type=3D"range"].svelte-

1kajgn1.svelte-1kajgn1::-webkit-slider-runnabl= e-track { background: linear-gradient(to right,var(--slider-color)

var(--ra= nge\_progress),var(--neutral-200) var(--range\_progress)); } input[type=3D"number"].svelte-1kajgn1.svelte-

1kajgn1 { display: block; outl= ine: none; border: 1px solid var(--input-border-color); border-radius: var(= --radius-sm);

background: var(--input-background-fill); padding: var(--size= -2) var(--size-3); height: var(--size-8); color: var(--body-

text-color); fo= nt-size: var(--input-text-size); line-height: var(--line-sm); text-align: c= enter; min-width: var(--size-16);

transition: border-color 0.15s ease-in-ou= t; } input[type=3D"number"].svelte-1kajgn1.svelte-1kajgn1:focus { box-

shadow: no= ne; border-width: 2px; } input.svelte-1kajgn1.svelte-1kajgn1:disabled, input[disabled].svelte-1kajgn=

1.svelte-1kajgn1 { -webkit-text-fill-color: var(--body-text-color); opacity= : 1; cursor: not-allowed; } input.svelte-

1kajgn1.svelte-1kajgn1::placeholder { color: var(--input-place= holder-color); } input[type=3D"range"][disabled].svelte-

1kajgn1.svelte-1kajgn1 { opacity: 0.= 6; } .min\_value.svelte-1kajgn1.svelte-1kajgn1, .max\_value.svelte-1kajgn1.svelte-

= 1kajgn1 { font-size: var(--text-sm); color: var(--body-text-color-subdued);= } .min\_value.svelte-1kajgn1.svelte-

1kajgn1 { margin-right: var(--size-0-5); } .max\_value.svelte-1kajgn1.svelte-1kajgn1 { margin-left: var(--size-0-5); ma=

rgin-right: var(--size-0-5); } @media (max-width: 480px) { .min\_value.svelte-1kajgn1.svelte-

1kajgn1, .max\_value.svelte-1kajgn1.svelt= e-1kajgn1 { display: none; } } @media (max-width: 520px) { .head.svelte-

1kajgn1.svelte-1kajgn1 { gap: var(--size-3); } } @media (max-width: 420px) { .head.svelte-1kajgn1 .tab-like-

container.svelte-1kajgn1 { margin-bottom: = var(--size-4); } } .tab-like-container.svelte-1kajgn1.svelte-1kajgn1

{ display: flex; align-it= ems: stretch; border: 1px solid var(--input-border-color); border-radius: v= ar(--radius-sm);

overflow: hidden; height: var(--size-6); } input[type=3D"number"].svelte-1kajgn1.svelte-1kajgn1 { border: none;

border= -radius: 0px; padding: var(--size-1) var(--size-2); height: 100%; min-width= : var(--size-14); font-size:

var(--text-sm); } input[type=3D"number"].svelte-1kajgn1.svelte-1kajgn1:focus { box-shadow: in= set 0 0 0 1px

var(--color-accent); border-radius: 3px 0px 0px 3px; } .reset-button.svelte-1kajgn1.svelte-1kajgn1 { display: flex; align-

items: c= enter; justify-content: center; background: none; border-top: none; border-= right: none; border-bottom: none;

border-image: initial; border-left: 1px s= olid var(--input-border-color); cursor: pointer; font-size: var(--text-sm);= color:

var(--body-text-color); padding: 0 var(--size-2); min-width: var(--= size-6); transition: background-color 0.15s ease-in-

out; } .reset-button.svelte-1kajgn1.svelte-1kajgn1:hover:not(:disabled) { backgrou= nd-color: var(--background-fill-

secondary); } .reset-button.svelte-1kajgn1.svelte-1kajgn1:disabled { opacity: 0.5; cursor= : not-allowed; } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-D-fiIdl6.css @charset "utf-8";

label.svelte-1jkmaen.svelte-1jkmaen:not(.container), label.svelte-1jkmaen:n= ot(.container) > input.svelte-1jkmaen

{ height: 100%; border: none; } .container.svelte-1jkmaen > input.svelte-1jkmaen { border: var(--input-bord= er-width)

solid var(--input-border-color); border-radius: var(--input-radiu= s); } input[type=3D"number"].svelte-1jkmaen.svelte-

1jkmaen { display: block; posi= tion: relative; box-shadow: var(--input-shadow); background: var(--input-ba= ckground-

fill); padding: var(--input-padding); width: 100%; color: var(--bo= dy-text-color); font-size: var(--input-text-size); line-

height: var(--line-= sm); outline: none !important; } input.svelte-1jkmaen.svelte-1jkmaen:disabled { -webkit-text-fill-

color: var= (--body-text-color); opacity: 1; } input.svelte-1jkmaen.svelte-1jkmaen:focus { box-shadow: var(--input-

shadow-= focus); border-color: var(--input-border-color-focus); background: var(--in= put-background-fill-focus); }

input.svelte-1jkmaen.svelte-1jkmaen::placeholder { color: var(--input-place= holder-color); } input.svelte-

1jkmaen.svelte-1jkmaen:out-of-range { border: var(--input-bord= er-width) solid var(--error-border-color); }

div.validation-error.svelte-1jkmaen.svelte-1jkmaen { color: var(--error-ico= n-color); font-size: var(--font-sans); margin-

top: var(--spacing-sm); font-= weight: var(--weight-semibold); } label.container.svelte-1jkmaen input.validation-

error.svelte-1jkmaen { bord= er-color: transparent !important; box-shadow: 0 0 3px 1px var(--error-icon-=

color),var(--shadow-inset) !important; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-a-pQRFMj.css @charset "utf-8"; label.svelte-

14vb072.svelte-14vb072.svelte-14vb072 { display: flex; align-i= tems: center; transition: var(--button-transition); cursor:

pointer; color:= var(--checkbox-label-text-color); font-weight: var(--checkbox-label-text-w= eight); font-size:

var(--checkbox-label-text-size); line-height: var(--line= -md); } label.svelte-14vb072 > .svelte-14vb072 + .svelte-

14vb072 { margin-left: var= (--size-2); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072 { --ring-color: transpar=

ent; position: relative; box-shadow: var(--checkbox-shadow); border: 1px so= lid var(--checkbox-border-color); border-

radius: var(--checkbox-border-radi= us); background-color: var(--checkbox-background-color); line-height: var(-= -line-

sm); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:checked, input.svelte-14= vb072.svelte-14vb072.svelte-

14vb072:checked:hover, input.svelte-14vb072.sve= lte-14vb072.svelte-14vb072:checked:focus { background-image:

var(--checkbox= -check); background-color: var(--checkbox-background-color-selected); borde= r-color:

var(--checkbox-border-color-focus); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:checked:focus { backgrou=

nd-image: var(--checkbox-check); background-color: var(--checkbox-backgroun= d-color-selected); border-color:

var(--checkbox-border-color-focus); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:hover { border-color: va=

r(--checkbox-border-color-hover); background-color: var(--checkbox-backgrou= nd-color-hover); } input.svelte-

14vb072.svelte-14vb072.svelte-14vb072:focus { border-color: va= r(--checkbox-border-color-focus); background-color:

var(--checkbox-backgrou= nd-color-focus); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:indeterminate

{ backgrou= nd-image: none; background-color: var(--checkbox-background-color-selected)= ; border-color:

var(--checkbox-border-color-focus); position: relative; } input.svelte-14vb072.svelte-14vb072.svelte-

14vb072:indeterminate::after { c= ontent: ""; position: absolute; top: 50%; left: 50%; transform: translate(-= 50%, -

50%); width: 8px; height: 2px; background-color: rgb(255, 255, 255); = } input.svelte-14vb072.svelte-14vb072.svelte-

14vb072:indeterminate:hover { ba= ckground-color: var(--checkbox-background-color-selected); border-color: va=

r(--checkbox-border-color-hover); } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:indeterminate:focus { ba=

ckground-color: var(--checkbox-background-color-selected); border-color: va= r(--checkbox-border-color-focus); }

input[disabled].svelte-14vb072.svelte-14vb072.svelte-14vb072, .disabled.sve= lte-14vb072.svelte-14vb072.svelte-

14vb072 { cursor: not-allowed !important;= } input.svelte-14vb072.svelte-14vb072.svelte-14vb072:hover { cursor:

pointer;= } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type:

text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-

YcGCkygm.css @charset "utf-8"; span.svelte-1w6vloh { font-weight: var(--section-header-text-weight); font-= size:

var(--section-header-text-size); } .label-wrap.svelte-1w6vloh { display: flex; justify-content: space-between;= cursor:

pointer; width: var(--size-full); color: var(--accordion-text-colo= r); } .label-wrap.open.svelte-1w6vloh { margin-bottom:

var(--size-2); } .icon.svelte-1w6vloh { transition: 0.15s; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-C-vZ3ifd.css @charset "utf-8"; .label-clear-

button.svelte-1rvzbk6.svelte-1rvzbk6 { color: var(--body-text-= color-subdued); position: relative; left: -3px; } .label-

clear-button.svelte-1rvzbk6.svelte-1rvzbk6:hover { color: var(--body= -text-color); } .file-preview.svelte-1rvzbk6.svelte-

1rvzbk6 { table-layout: fixed; width: v= ar(--size-full); max-height: var(--size-60); overflow-y: auto; margin-top: =

var(--size-1); color: var(--body-text-color); } .file-preview-holder.svelte-1rvzbk6.svelte-1rvzbk6 { overflow:

auto; } .file.svelte-1rvzbk6.svelte-1rvzbk6 { display: flex; width: var(--size-full= ); } .file.svelte-1rvzbk6 > .svelte-

1rvzbk6 { padding: var(--size-1) var(--size-= 2-5); } .filename.svelte-1rvzbk6.svelte-1rvzbk6 { flex-grow: 1; display:

flex; over= flow: hidden; } .filename.svelte-1rvzbk6 .stem.svelte-1rvzbk6 { overflow: hidden; text-over= flow: ellipsis;

white-space: nowrap; } .filename.svelte-1rvzbk6 .ext.svelte-1rvzbk6 { white-space: nowrap; } .download.svelte-

1rvzbk6.svelte-1rvzbk6 { min-width: 8rem; width: 10%; whit= e-space: nowrap; text-align: right; } .download.svelte-

1rvzbk6.svelte-1rvzbk6:hover { text-decoration: underline;= } .download.svelte-1rvzbk6 > a { color: var(--link-text-

color); } .download.svelte-1rvzbk6 > a:hover { color: var(--link-text-color-hover); } .download.svelte-1rvzbk6 >

a:visited { color: var(--link-text-color-visited= ); } .download.svelte-1rvzbk6 > a:active { color: var(--link-text-color-

active);= } .selectable.svelte-1rvzbk6.svelte-1rvzbk6 { cursor: pointer; } tbody.svelte-1rvzbk6 > tr.svelte-1rvzbk6:nth-

child(2n) { background: var(--= block-background-fill); } tbody.svelte-1rvzbk6 > tr.svelte-1rvzbk6:nth-child(2n+1)

{ background: var(= --table-odd-background-fill); } .drag-handle.svelte-1rvzbk6.svelte-1rvzbk6 { cursor: grab; color:

var(--bod= y-text-color-subdued); padding-right: var(--size-2); user-select: none; } .dragging.svelte-1rvzbk6.svelte-

1rvzbk6 { opacity: 0.5; cursor: grabbing; } .drop-target.svelte-1rvzbk6.svelte-1rvzbk6 { border-top: 2px solid var(--co=

lor-accent); } tr:last-child.drop-target[data-drop-target=3D"before"].svelte-1rvzbk6.svelt= e-1rvzbk6 { border-top: 2px

solid var(--color-accent); border-bottom: none;= } tr:last-child.drop-target[data-drop-target=3D"after"].svelte-

1rvzbk6.svelte= -1rvzbk6 { border-top: none; border-bottom: 2px solid var(--color-accent); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Example-BjJvNKVN.css @charset "utf-8"; div.svelte-

rgtszb { overflow: hidden; text-overflow: ellipsis; white-space:= nowrap; } .gallery.svelte-rgtszb { display: flex; align-

items: center; cursor: pointe= r; padding: var(--size-1) var(--size-2); text-align: left; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-D812IjdW.css @charset "utf-8"; label.svelte-

1ae7ssi.svelte-1ae7ssi { display: block; width: 100%; } input.svelte-1ae7ssi.svelte-1ae7ssi, textarea.svelte-1ae7ssi.svelte-

1ae7ssi= { flex-grow: 1; margin-top: 0px; margin-bottom: 0px; resize: none; z-index= : 1; display: block; position:

relative; background: var(--input-background= -fill); padding: var(--input-padding); width: 100%; color: var(--body-text-

= color); font-weight: var(--input-text-weight); font-size: var(--input-text-= size); line-height: var(--line-sm); border:

none; outline: none !important;= } textarea.no-label.svelte-1ae7ssi.svelte-1ae7ssi { padding-top: 5px; padding= -bottom:

5px; } label.show\_textbox\_border.svelte-1ae7ssi input.svelte-1ae7ssi, label.show\_t= extbox\_border.svelte-1ae7ssi

textarea.svelte-1ae7ssi { box-shadow: var(--in= put-shadow); } label.svelte-1ae7ssi.svelte-1ae7ssi:not(.container),

label.svelte-1ae7ssi:n= ot(.container) input.svelte-1ae7ssi, label.svelte-1ae7ssi:not(.container) t= extarea.svelte-1ae7ssi

{ height: 100%; } label.container.show\_textbox\_border.svelte-1ae7ssi input.svelte-1ae7ssi, la=

bel.container.show\_textbox\_border.svelte-1ae7ssi textarea.svelte-1ae7ssi { = border: var(--input-border-width) solid

var(--input-border-color); border-r= adius: var(--input-radius); } input.svelte-1ae7ssi.svelte-1ae7ssi:disabled,

textarea.svelte-1ae7ssi.svelt= e-1ae7ssi:disabled { opacity: 1; } label.container.show\_textbox\_border.svelte-1ae7ssi

input.svelte-1ae7ssi:foc= us, label.container.show\_textbox\_border.svelte-1ae7ssi textarea.svelte-1ae7= ssi:focus { box-

shadow: var(--input-shadow-focus); border-color: var(--inpu= t-border-color-focus); background: var(--input-

background-fill-focus); } input.svelte-1ae7ssi.svelte-1ae7ssi::placeholder, textarea.svelte-1ae7ssi.s= velte-

1ae7ssi::placeholder { color: var(--input-placeholder-color); } .input-container.svelte-1ae7ssi.svelte-1ae7ssi { display:

flex; position: r= elative; align-items: flex-end; } .submit-button.svelte-1ae7ssi.svelte-1ae7ssi, .stop-button.svelte-

1ae7ssi.s= velte-1ae7ssi { border: none; text-align: center; text-decoration: none; fo= nt-size: 14px; cursor: pointer;

border-radius: 15px; min-width: 30px; heigh= t: 30px; flex-shrink: 0; display: flex; justify-content: center; align-item= s:

center; z-index: var(--layer-1); } .stop-button.svelte-1ae7ssi.svelte-1ae7ssi, .submit-button.svelte-1ae7ssi.s= velte-

1ae7ssi { background: var(--button-secondary-background-fill); color:= var(--button-secondary-text-color); } .stop-

button.svelte-1ae7ssi.svelte-1ae7ssi:hover, .submit-button.svelte-1ae= 7ssi.svelte-1ae7ssi:hover { background:

var(--button-secondary-background-f= ill-hover); } .stop-button.svelte-1ae7ssi.svelte-1ae7ssi:disabled, .submit-

button.svelte-= 1ae7ssi.svelte-1ae7ssi:disabled { background: var(--button-secondary-backgr= ound-fill); cursor:

pointer; } .stop-button.svelte-1ae7ssi.svelte-1ae7ssi:active, .submit-button.svelte-1a= e7ssi.svelte-1ae7ssi:active { box-

shadow: var(--button-shadow-active); } .submit-button.svelte-1ae7ssi svg { height: 22px; width: 22px; } .stop-

button.svelte-1ae7ssi svg { height: 16px; width: 16px; } .padded-button.svelte-1ae7ssi.svelte-1ae7ssi { padding: 0px

10px; } div.validation-error.svelte-1ae7ssi.svelte-1ae7ssi { color: var(--error-ico= n-color); font-size: var(--font-sans);

margin-top: var(--spacing-sm); font-= weight: var(--weight-semibold); } label.container.svelte-1ae7ssi input.validation-

error.svelte-1ae7ssi, label= .container.svelte-1ae7ssi textarea.validation-error.svelte-1ae7ssi { border= -color:

transparent !important; box-shadow: 0 0 3px 1px var(--error-icon-co= lor),var(--shadow-inset) !important; } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Example-DRnq2kzJ.css @charset

"utf-8"; .gallery.svelte-1oitfqa { padding: var(--size-1) var(--size-2); } div.svelte-1oitfqa { overflow: hidden; white-

space: nowrap; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type:

text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/InteractiveAudio-

DkbXm\_Wx.css @charset "utf-8"; .mic-select.svelte-1ya9x7a { height: var(--size-8); background: var(--block= -

background-fill); padding: 0px var(--spacing-xxl); border-radius: var(--bu= tton-large-radius); font-size: var(--text-md);

border: 1px solid var(--bloc= k-border-color); gap: var(--size-1); } select.svelte-1ya9x7a { text-overflow: ellipsis; max-

width: var(--size-40);= } @media (max-width: 375px) { select.svelte-1ya9x7a { width: 100%; } } .controls.svelte-

1oiuk2f { display: flex; align-items: center; justify-cont= ent: space-between; flex-wrap: wrap; } .wrapper.svelte-1oiuk2f

{ display: flex; align-items: center; flex-wrap: wr= ap; } .record.svelte-1oiuk2f { margin-right: var(--spacing-

md); } .stop-button-paused.svelte-1oiuk2f { display: none; height: var(--size-8); = width: var(--size-20); background-

color: var(--block-background-fill); bord= er-radius: var(--button-large-radius); align-items: center; border: 1px sol= id

var(--block-border-color); margin: var(--size-1) var(--size-1) 0 0; } .stop-button-paused.svelte-1oiuk2f::before { content:

""; height: var(--siz= e-4); width: var(--size-4); border-radius: var(--radius-full); background: = var(--primary-600);

margin: 0 var(--spacing-xl); } .stop-button.svelte-1oiuk2f::before { content: ""; height: var(--size-4); w= idth: var(--size-

4); border-radius: var(--radius-full); background: var(--p= rimary-600); margin: 0 var(--spacing-xl); animation: 1.8s ease

0s infinite = normal none running svelte-1oiuk2f-scaling; } .stop-button.svelte-1oiuk2f { display: none; height:

var(--size-8); width: = var(--size-20); background-color: var(--block-background-fill); border-radi= us: var(--button-

large-radius); align-items: center; border: 1px solid var(= --primary-600); margin: var(--size-1) var(--size-1) 0

0; } .record-button.svelte-1oiuk2f::before { content: ""; height: var(--size-4);= width: var(--size-4); border-radius:

var(--radius-full); background: var(-= -primary-600); margin: 0 var(--spacing-xl); } .record-button.svelte-1oiuk2f

{ height: var(--size-8); width: var(--size-24= ); background-color: var(--block-background-fill); border-radius: var(--but=

ton-large-radius); display: flex; align-items: center; border: 1px solid va= r(--block-border-color); } .duration-

button.svelte-1oiuk2f { border-radius: var(--button-large-radius)= ; } .stop-button.svelte-1oiuk2f:disabled { cursor: not-

allowed; } .record-button.svelte-1oiuk2f:disabled { cursor: not-allowed; opacity: 0.5;= } @keyframes svelte-1oiuk2f-

scaling {=20 0% { background-color: var(--primary-600); scale: 1; } 50% { background-color: var(--primary-600);

scale: 1.2; } 100% { background-color: var(--primary-600); scale: 1; } } .pause-button.svelte-1oiuk2f { display: none;

height: var(--size-8); width:= var(--size-20); border: 1px solid var(--block-border-color); border-radius= : var(--button-

large-radius); padding: var(--spacing-md); margin: var(--siz= e-1) var(--size-1) 0 0; } .resume-button.svelte-1oiuk2f

{ display: none; height: var(--size-8); width= : var(--size-20); border: 1px solid var(--block-border-color); border-radiu=

s: var(--button-large-radius); padding: var(--spacing-xl); line-height: 1px= ; font-size: var(--text-md); margin: var(--size-

1) var(--size-1) 0 0; } .duration.svelte-1oiuk2f { display: flex; height: var(--size-8); width: var= (--size-20); border: 1px

solid var(--block-border-color); padding: var(--sp= acing-md); align-items: center; justify-content: center; margin:

var(--size= -1) var(--size-1) 0 0; } ::part(region) { border-radius: var(--radius-md); border-top-color: ; borde= r-top-

style: ; border-right-color: ; border-right-style: ; border-bottom-co= lor: ; border-bottom-style: ; border-left-color: ;

border-left-style: ; bor= der-image-source: ; border-image-slice: ; border-image-width: ; border-imag= e-outset: ; border-

image-repeat: ; background-color: unset; border-width: 1= px 3px; height: 98% !important; } ::part(region)::after

{ content: ""; position: absolute; top: 0px; left: 0p= x; width: 100%; height: 100%; background: var(--trim-region-color);

opacity= : 0.2; border-radius: var(--radius-md); } ::part(region-handle) { border: none; width: 5px !

important; } .microphone.svelte-9n45fh { width: 100%; display: none; } .component-wrapper.svelte-9n45fh { padding:

var(--size-3); width: 100%; } .timestamps.svelte-9n45fh { display: flex; justify-content: space-between; = align-items:

center; width: 100%; padding: var(--size-1) 0; margin: var(--s= pacing-md) 0; } .time.svelte-9n45fh, .duration.svelte-

9n45fh { color: var(--neutral-400); } .trim-duration.svelte-9n45fh { color: var(--color-accent); margin-right: va=

r(--spacing-sm); } .controls.svelte-1fz19cj { display: flex; align-items: center; justify-cont= ent: space-between; flex-

wrap: wrap; } .mic-wrap.svelte-1fz19cj { display: block; align-items: center; margin: var= (--spacing-xl); } .icon.svelte-

1fz19cj { width: var(--size-4); height: var(--size-4); fill: v= ar(--primary-600); stroke: var(--primary-600); } .stop-button-

paused.svelte-1fz19cj { display: none; height: var(--size-8); = width: var(--size-20); background-color: var(--block-

background-fill); bord= er-radius: var(--button-large-radius); align-items: center; border: 1px sol= id var(--block-border-

color); margin-right: 5px; } .stop-button-paused.svelte-1fz19cj::before { content: ""; height: var(--siz= e-4); width:

var(--size-4); border-radius: var(--radius-full); background: = var(--primary-600); margin: 0 var(--spacing-xl); } .stop-

button.svelte-1fz19cj::before { content: ""; height: var(--size-4); w= idth: var(--size-4); border-radius: var(--radius-full);

background: var(--p= rimary-600); margin: 0 var(--spacing-xl); animation: 1.8s ease 0s infinite = normal none running

svelte-1fz19cj-scaling; } .stop-button.svelte-1fz19cj { height: var(--size-8); width: var(--size-20);= background-color:

var(--block-background-fill); border-radius: var(--butto= n-large-radius); align-items: center; border: 1px solid

var(--primary-600);= margin-right: 5px; display: flex; } .spinner-button.svelte-1fz19cj { height: var(--size-8); width:

var(--size-2= 4); background-color: var(--block-background-fill); border-radius: var(--ra= dius-3xl); align-items: center;

border: 1px solid var(--primary-600); margi= n: 0 var(--spacing-xl); display: flex; justify-content: space-

evenly; } .record-button.svelte-1fz19cj::before { content: ""; height: var(--size-4);= width: var(--size-4); border-radius:

var(--radius-full); background: var(-= -primary-600); margin: 0 var(--spacing-xl); } .record-button.svelte-1fz19cj

{ height: var(--size-8); width: var(--size-24= ); background-color: var(--block-background-fill); border-radius: var(--but=

ton-large-radius); display: flex; align-items: center; border: 1px solid va= r(--block-border-color); } @keyframes svelte-

1fz19cj-scaling {=20 0% { background-color: var(--primary-600); scale: 1; } 50% { background-color: var(--primary-

600); scale: 1.2; } 100% { background-color: var(--primary-600); scale: 1; } } .audio-container.svelte-1ud6e7m { height:

calc(var(--size-full) - var(--siz= e-6)); display: flex; flex-direction: column; justify-content: space-betwee= n; } .audio-

container.compact-audio.svelte-1ud6e7m { margin-top: calc(var(--size= -8) \* -1); height: auto; padding: 0px; gap:

var(--size-2); min-height: var(= --size-5); } .compact-audio.svelte-1ud6e7m .audio-player { padding: 0px; } .compact-

audio.svelte-1ud6e7m .controls { gap: 0px; padding: 0px; } .compact-audio.svelte-1ud6e7m .waveform-container

{ height: var(--size-12) = !important; } .compact-audio.svelte-1ud6e7m .player-container { min-height: unset; height= :

auto; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css

Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/ModifyUpload-

REHVxGq-.css @charset "utf-8"; .wrap.svelte-1vsfomn.svelte-1vsfomn { overflow-y: auto; transition: opacity= 0.5s

ease-in-out; background: var(--block-background-fill); position: rela= tive; display: flex; flex-direction: column; align-

items: center; justify-c= ontent: center; min-height: var(--size-40); width: var(--size-full); } .wrap.svelte-1vsfomn.svelte-

1vsfomn::after { content: ""; position: absolut= e; top: 0px; left: 0px; width: var(--upload-progress-width); height: 100%;

= transition: 0.5s ease-in-out; z-index: 1; } .uploading.svelte-1vsfomn.svelte-1vsfomn { font-size: var(--text-lg); font-=

family: var(--font); z-index: 2; } .file-name.svelte-1vsfomn.svelte-1vsfomn { margin: var(--spacing-md); font-= size:

var(--text-lg); color: var(--body-text-color-subdued); } .file.svelte-1vsfomn.svelte-1vsfomn { font-size: var(--text-md); z-

index: 2= ; display: flex; align-items: center; } .file.svelte-1vsfomn progress.svelte-1vsfomn { display: inline; height:

var= (--size-1); width: 100%; transition: 0.5s ease-in-out; color: var(--color-a= ccent); border: none; } .file.svelte-

1vsfomn progress[value].svelte-1vsfomn::-webkit-progress-value= { background-color: var(--color-accent); border-

radius: 20px; } .file.svelte-1vsfomn progress[value].svelte-1vsfomn::-webkit-progress-bar {= background-color:

var(--border-color-accent); border-radius: 20px; } .progress-bar.svelte-1vsfomn.svelte-1vsfomn { width: 14px; height:

14px; bo= rder-radius: 50%; background: radial-gradient(closest-side,var(--block-back= ground-fill) 64%,transparent

53% 100%),conic-gradient(var(--color-accent) v= ar(--upload-progress-width),var(--border-color-accent) 0); transition:

0.5s= ease-in-out; } button.svelte-edrmkl { cursor: pointer; width: var(--size-full); } .center.svelte-edrmkl { display: flex;

justify-content: center; } .flex.svelte-edrmkl { display: flex; flex-direction: column; justify-conten= t: center; align-items:

center; } .hidden.svelte-edrmkl { display: none; position: absolute; flex-grow: 0; } .hidden.svelte-edrmkl svg { display:

none; } .disable\_click.svelte-edrmkl { cursor: default; } .icon-mode.svelte-edrmkl { width: var(--size-4); height:

var(--size-4); pad= ding: 0px; min-height: 0px; border-radius: var(--radius-circle); position: = absolute !

important; } .icon-mode.svelte-edrmkl svg { width: var(--size-4); height: var(--size-4);= } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/AudioPlayer-Bioz4LGv.css @charset "utf-8"; .volume-

slider.svelte-wuo8j5 { appearance: none; width: var(--size-20); acc= ent-color: var(--color-accent); height: 4px; cursor:

pointer; outline: none= ; border-radius: 15px; background-color: var(--neutral-400); } input[type=3D"range"].svelte-

wuo8j5::-webkit-slider-thumb { appearance: non= e; height: 15px; width: 15px; background-color: var(--color-accent);

border= -radius: 50%; border: none; transition: 0.2s ease-in-out; } .settings-wrapper.svelte-1joklbr.svelte-1joklbr

{ display: flex; justify-se= lf: self-end; align-items: center; grid-area: editing; } .text-button.svelte-1joklbr.svelte-1joklbr

{ border: 1px solid var(--neutra= l-400); border-radius: var(--radius-sm); font-size: var(--size-3); text-ali= gn: center;

color: var(--neutral-400); height: var(--size-5); font-weight: = 700; padding: 0px 5px; margin-left: 5px; } .text-

button.svelte-1joklbr.svelte-1joklbr:hover, .text-button.svelte-1jokl= br.svelte-1joklbr:focus { color: var(--color-accent);

border-color: var(--c= olor-accent); } .controls.svelte-1joklbr.svelte-1joklbr { display: grid; grid-template-colu= mns: 1fr

1fr 1fr; grid-template-areas: "controls playback editing"; margin-= top: 5px; align-items: center; position: relative; flex-

wrap: wrap; justify= -content: space-between; } .controls.svelte-1joklbr div.svelte-1joklbr { margin: var(--size-1) 0; }

@media (max-width: 600px) { .controls.svelte-1joklbr.svelte-1joklbr { grid-template: "playback playba= ck" "controls

editing" / 1fr 1fr; } } @media (max-width: 319px) { .controls.svelte-1joklbr.svelte-1joklbr { overflow-x:

scroll; } } .hidden.svelte-1joklbr.svelte-1joklbr { display: none; } .control-wrapper.svelte-1joklbr.svelte-1joklbr

{ display: flex; justify-sel= f: self-start; align-items: center; justify-content: space-between; grid-ar= ea:

controls; } .action.svelte-1joklbr.svelte-1joklbr { width: var(--size-5); color: var(--= neutral-400); margin-left:

var(--spacing-md); } .icon.svelte-1joklbr.svelte-1joklbr:hover, .icon.svelte-1joklbr.svelte-1jok= lbr:focus { color:

var(--color-accent); } .play-pause-wrapper.svelte-1joklbr.svelte-1joklbr { display: flex; justify-= self: center; grid-area:

playback; } .cc-button.svelte-1joklbr.svelte-1joklbr { width: var(--size-8); } @media (max-width: 600px) { .play-pause-

wrapper.svelte-1joklbr.svelte-1joklbr { margin: var(--spacing= -md); } } .playback.svelte-1joklbr.svelte-1joklbr { border:

1px solid var(--neutral-4= 00); border-radius: var(--radius-sm); width: 5.5ch; font-size: var(--size-3= ); text-align: center;

color: var(--neutral-400); height: var(--size-5); fo= nt-weight: 700; } .playback.svelte-1joklbr.svelte-

1joklbr:hover, .playback.svelte-1joklbr.sve= lte-1joklbr:focus { color: var(--color-accent); border-color: var(--color-a=

ccent); } .rewind.svelte-1joklbr.svelte-1joklbr, .skip.svelte-1joklbr.svelte-1joklbr = { margin: 0px 10px; color:

var(--neutral-400); } .play-pause-button.svelte-1joklbr.svelte-1joklbr { width: var(--size-8); di= splay: flex; align-items:

center; justify-content: center; color: var(--neu= tral-400); fill: var(--neutral-400); } .volume.svelte-1joklbr.svelte-1joklbr

{ position: relative; display: flex; = justify-content: center; margin-right: var(--spacing-xl); width: var(--size= -

5); } .component-wrapper.svelte-1vm10g7 { padding: var(--size-3); width: 100%; } ::part(wrapper) { margin-bottom:

var(--size-2); } .timestamps.svelte-1vm10g7 { display: flex; justify-content: space-between;= align-items: center; width:

100%; padding: var(--size-1) 0; } #time.svelte-1vm10g7, #duration.svelte-1vm10g7 { color: var(--neutral-400);= }

#trim-duration.svelte-1vm10g7 { color: var(--color-accent); margin-right: v= ar(--spacing-sm); } .waveform-

container.svelte-1vm10g7 { display: flex; align-items: center; ju= stify-content: center; width: var(--size-full); }

#waveform.svelte-1vm10g7 { width: 100%; height: 100%; position: relative; } .standard-player.svelte-1vm10g7 { width:

100%; padding: var(--size-2); } .subtitle-display.svelte-1vm10g7 { color: var(--text-secondary); font-size:= var(--text-

lg); text-align: center; max-width: 600px; line-height: 1.3; mi= n-height: var(--size-4); font-family: var(--font-sans);

font-weight: 400; m= argin: var(--size-2) auto; padding: var(--size-1) var(--size-2); border-rad= ius: 2px; transition:

opacity 0.2s ease-in-out; } .hidden.svelte-1vm10g7, .subtitle-display.svelte-1vm10g7:empty { display: n= one; } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-C8GNV2pm.css @charset

"utf-8"; div.svelte-5cspr0 { display: flex; flex-direction: column; position: relati= ve; border: none; border-radius:

var(--radius-sm); width: 100%; box-sizing:= border-box; } .grow-children.svelte-5cspr0 > .column > .column { flex-

grow: 1; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type:

text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-Bp6-

wsMo.css @charset "utf-8"; div.svelte-bnzux8 { display: flex; position: relative; flex-direction: colu= mn; gap:

var(--layout-gap); } div.svelte-bnzux8 > \*, div.svelte-bnzux8 > .form > \* { width: var(--size-fu= ll); } .hide.svelte-

bnzux8 { display: none; } .compact.svelte-bnzux8 > \*, .compact.svelte-bnzux8 .box { border-radius: 0p=

x; } .compact.svelte-bnzux8, .panel.svelte-bnzux8 { border: solid var(--panel-bo= rder-width) var(--panel-border-color);

border-radius: var(--container-radiu= s); background: var(--panel-background-fill); padding: var(--spacing-lg); } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-DqCgYdoJ.css @charset "utf-

8"; .tabs.svelte-i00v67.svelte-i00v67 { position: relative; display: flex; flex= -direction: column; gap: var(--layout-

gap); } .hide.svelte-i00v67.svelte-i00v67 { display: none; } .hidden.svelte-i00v67.svelte-i00v67 { display: none !

important; } .tab-wrapper.svelte-i00v67.svelte-i00v67 { display: flex; align-items: cent= er; justify-content: space-

between; position: relative; height: var(--size-= 8); padding-bottom: var(--size-2); } .tab-container.svelte-i00v67.svelte-

i00v67 { display: flex; align-items: ce= nter; width: 100%; position: relative; overflow: hidden; height: var(--size= -

8); } .tab-container.svelte-i00v67.svelte-i00v67::after { content: ""; position: = absolute; bottom: 0px; left: 0px; right:

0px; height: 1px; background-color= : var(--border-color-primary); } .overflow-menu.svelte-i00v67.svelte-i00v67 { flex-

shrink: 0; margin-left: v= ar(--size-2); } button.svelte-i00v67.svelte-i00v67 { margin-bottom: 0px; border: none; bord=

er-radius: 0px; padding: 0 var(--size-4); color: var(--body-text-color); fo= nt-weight: var(--section-header-text-weight);

font-size: var(--section-head= er-text-size); background-color: transparent; height: 100%; display: flex; = align-items:

center; white-space: nowrap; position: relative; } button.svelte-i00v67.svelte-i00v67:disabled { opacity: 0.5; cursor: not-

all= owed; } button.svelte-i00v67.svelte-i00v67:hover:not(:disabled):not(.selected) { ba= ckground-color:

var(--background-fill-secondary); color: var(--body-text-co= lor); } .selected.svelte-i00v67.svelte-i00v67 { background-

color: transparent; colo= r: var(--color-accent); position: relative; } .selected.svelte-i00v67.svelte-i00v67::after { content:

""; position: absol= ute; bottom: 0px; left: 0px; width: 100%; height: 2px; background-color: va= r(--color-accent);

animation: 0.2s ease-out 0s 1 normal forwards running sv= elte-i00v67-fade-grow; transform-origin: center center; z-

index: 1; } @keyframes svelte-i00v67-fade-grow {=20 0% { opacity: 0; transform: scaleX(0.8); } 100% { opacity: 1;

transform: scaleX(1); } } .overflow-dropdown.svelte-i00v67.svelte-i00v67 { position: absolute; top: c= alc(100% +

var(--size-2)); right: 0px; background-color: var(--background-f= ill-primary); border: 1px solid var(--border-color-

primary); border-radius:= var(--radius-sm); z-index: var(--layer-5); box-shadow: rgba(0, 0, 0, 0.1) = 0px 2px 5px;

padding: var(--size-2); min-width: 150px; width: max-content; = } .overflow-dropdown.svelte-i00v67 button.svelte-

i00v67 { display: block; wid= th: 100%; text-align: left; padding: var(--size-2); white-space: nowrap; ov= erflow:

hidden; text-overflow: ellipsis; } .overflow-menu.svelte-i00v67 > button.svelte-i00v67 { padding: var(--size-1= )

var(--size-2); min-width: auto; border: 1px solid var(--border-color-prim= ary); border-radius: var(--radius-sm); display:

flex; align-items: center; = justify-content: center; } .overflow-menu.svelte-i00v67 > button.svelte-i00v67:hover

{ background-colo= r: var(--background-fill-secondary); } .overflow-menu.svelte-i00v67 svg { width: 16px; height:

16px; } .overflow-item-selected.svelte-i00v67 svg { color: var(--color-accent); } .visually-hidden.svelte-i00v67.svelte-

i00v67 { position: absolute; width: 1= px; height: 1px; padding: 0px; margin: -1px; overflow: hidden; clip: rect(0= px,

0px, 0px, 0px); white-space: nowrap; border: 0px; } .stepper.svelte-phhp3p.svelte-phhp3p { position: relative; display:

flex; f= lex-direction: column; gap: var(--layout-gap); } .compact.stepper.svelte-phhp3p.svelte-phhp3p { gap:

0px; } .hide.svelte-phhp3p.svelte-phhp3p { display: none; } .stepper-wrapper.svelte-phhp3p.svelte-phhp3p { display:

flex; align-items: = center; position: relative; padding-top: var(--size-4); padding-bottom: cal= c(var(--label-height) +

var(--size-4)); } .compact.svelte-phhp3p .stepper-wrapper.svelte-phhp3p { padding-top: var(--= size-2); padding-bottom:

var(--size-6); } .stepper-container.svelte-phhp3p.svelte-phhp3p { display: flex; justify-con= tent: space-between; align-

items: flex-start; width: 100%; position: relati= ve; padding: var(--size-2); gap: var(--size-1); } .compact.svelte-

phhp3p .stepper-container.svelte-phhp3p { justify-content: = center; gap: 2px; padding: 0px; } .step-item.svelte-

phhp3p.svelte-phhp3p { display: flex; align-items: center= ; justify-content: center; flex: 1 1 0px; position:

relative; } .compact.svelte-phhp3p .step-item.svelte-phhp3p { width: 100%; } .step-button.svelte-phhp3p.svelte-phhp3p

{ display: flex; flex-direction: c= olumn; align-items: center; justify-content: center; gap: var(--size-1); bo= rder: none;

background: transparent; cursor: pointer; border-radius: var(--= radius-md); transition: background-color 0.2s; font-size:

var(--text-sm); c= olor: var(--body-text-color-subdued); white-space: nowrap; z-index: 1; posi= tion:

relative; } .compact.svelte-phhp3p .step-button.svelte-phhp3p { padding: 0px; width: 10= 0%; border:

none; } .compact.svelte-phhp3p .step-number.svelte-phhp3p { height: 10px; width: 10= 0%; border-radius: 0px; border:

none; } .step-button.svelte-phhp3p.svelte-phhp3p:hover:not(:disabled) { background-= color: var(--background-fill-

secondary); } .step-button.svelte-phhp3p.svelte-phhp3p:disabled { cursor: not-allowed; op= acity: 0.5; } .step-

button.active.svelte-phhp3p.svelte-phhp3p, .step-button.completed.sve= lte-phhp3p.svelte-phhp3p { color: var(--body-

text-color); } .step-button.pending.svelte-phhp3p.svelte-phhp3p { color: var(--body-text-c= olor-subdued); } .step-

number.svelte-phhp3p.svelte-phhp3p { display: flex; align-items: cent= er; justify-content: center; width: 32px; height:

32px; border-radius: 50%;= font-size: var(--text-sm); font-weight: var(--weight-semibold); transition= : background-

color 0.2s; flex-shrink: 0; } .active.svelte-phhp3p .step-number.svelte-phhp3p { background-color: var(--= color-accent);

color: rgb(255, 255, 255); box-shadow: 0 0 0 4px rgba(var(--= color-accent-rgb),.1); } .completed.svelte-phhp3p .step-

number.svelte-phhp3p { background-color: var= (--color-accent); color: rgb(255, 255, 255); } .compact.svelte-

phhp3p .completed .step-number.svelte-phhp3p { color: trans= parent; } .compact.svelte-phhp3p .pending .step-

number.svelte-phhp3p { color: transpa= rent; background-color: var(--body-text-color-subdued); } .compact.svelte-

phhp3p .active .step-number.svelte-phhp3p { color: transpar= ent; } .pending.svelte-phhp3p .step-number.svelte-phhp3p

{ background-color: var(-= -button-secondary-background-fill); color: var(--button-secondary-text-colo=

r); } .compact.svelte-phhp3p .step-item:last-child .step-number.svelte-phhp3p { b= order-top-right-radius: var(--radius-

xs); border-bottom-right-radius: var(-= -radius-xs); } .compact.svelte-phhp3p .step-item:first-child .step-number.svelte-

phhp3p { = border-top-left-radius: var(--radius-xs); border-bottom-left-radius: var(--= radius-xs); } .step-label.svelte-

phhp3p.svelte-phhp3p { font-size: var(--text-md); line-h= eight: 1.2; text-align: center; max-width: 120px; position:

absolute; botto= m: -20px; display: none; } .step-label.visible.svelte-phhp3p.svelte-phhp3p { display: block; } .step-

connector.svelte-phhp3p.svelte-phhp3p { width: 100%; height: 2px; bac= kground-color: var(--border-color-primary);

transition: background-color 0.= 3s; z-index: 0; transform: translateY(15px); } .step-connector.completed.svelte-

phhp3p.svelte-phhp3p { background-color: v= ar(--color-accent); } .dark .pending.svelte-phhp3p .step-number.svelte-

phhp3p { background-color:= var(--neutral-800); color: var(--neutral-400); border-color: var(--neutral= -

600); } .dark .step-connector.svelte-phhp3p.svelte-phhp3p { background-color: var(-= -neutral-600); } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-DMXFGchm.css @charset

"utf-8"; div.svelte-1vd8eap { display: flex; flex-direction: inherit; flex-wrap: wra= p; gap: var(--form-gap-width); box-

shadow: var(--block-shadow); border: var= (--block-border-width) solid var(--block-border-color); border-radius:

var(= --block-radius); background: var(--border-color-primary); overflow-y: hidde= n; } div.svelte-1vd8eap .block

{ box-shadow: none !important; border-width: 0px = !important; border-radius: 0px !important; } .hidden.svelte-1vd8eap

{ display: none; } .hidden-css.svelte-1vd8eap { display: none !important; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-CZxO08Kq.css @charset "utf-8"; label.svelte-

1bx8sav.svelte-1bx8sav.svelte-1bx8sav { display: flex; align-i= tems: center; transition: var(--button-transition); cursor:

pointer; box-sh= adow: var(--checkbox-label-shadow); border: var(--checkbox-label-border-wid= th) solid

var(--checkbox-label-border-color); border-radius: var(--checkbox= -border-radius); background: var(--checkbox-label-

background-fill); padding= : var(--checkbox-label-padding); color: var(--checkbox-label-text-color); f= ont-weight:

var(--checkbox-label-text-weight); font-size: var(--checkbox-la= bel-text-size); line-height: var(--line-md); }

label.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav:hover { background: var(= --checkbox-label-background-fill-

hover); } label.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav:focus { background: var(= --checkbox-label-background-

fill-focus); } label.selected.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav { background: v= ar(--checkbox-label-

background-fill-selected); color: var(--checkbox-label-= text-color-selected); border-color: var(--checkbox-label-border-

color-selec= ted); } label.svelte-1bx8sav > .svelte-1bx8sav + .svelte-1bx8sav { margin-left: var= (--size-2); }

label.rtl.svelte-1bx8sav > .svelte-1bx8sav + .svelte-1bx8sav { margin-left:= 0px; margin-right: var(--size-2); }

input.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav { --ring-color: transpar= ent; position: relative; box-shadow:

var(--checkbox-shadow); border: var(--= checkbox-border-width) solid var(--checkbox-border-color); border-radius: v=

ar(--radius-full); background-color: var(--checkbox-background-color); line= -height: var(--line-sm); } input.svelte-

1bx8sav.svelte-1bx8sav.svelte-1bx8sav:checked, input.svelte-1b= x8sav.svelte-1bx8sav.svelte-1bx8sav:checked:hover

{ border-color: var(--che= ckbox-border-color-selected); background-image: var(--radio-circle); backgr= ound-color:

var(--checkbox-background-color-selected); } input.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav:checked::after

{ content= : ""; position: absolute; top: 50%; left: 50%; transform: translate(-50%, -= 50%); border-radius: 50%;

background-color: rgb(255, 255, 255); } input.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav:hover { border-color: va=

r(--checkbox-border-color-hover); background-color: var(--checkbox-backgrou= nd-color-hover); } input.svelte-

1bx8sav.svelte-1bx8sav.svelte-1bx8sav:focus { border-color: va= r(--checkbox-border-color-focus); background-color:

var(--checkbox-backgrou= nd-color-focus); } input.svelte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav:checked:focus

{ border-c= olor: var(--checkbox-border-color-focus); background-image: var(--radio-cir= cle); background-color:

var(--checkbox-background-color-selected); } input[disabled].svelte-1bx8sav.svelte-1bx8sav.svelte-

1bx8sav, .disabled.sve= lte-1bx8sav.svelte-1bx8sav.svelte-1bx8sav { cursor: not-allowed; } .wrap.svelte-1kzox3m

{ display: flex; flex-wrap: wrap; gap: var(--checkbox-= label-gap); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Index-Ci4eVLjY.css @charset "utf-8"; div.svelte-

1xp0cw7 { display: flex; flex-wrap: wrap; gap: var(--layout-gap)= ; width: var(--size-full); position:

relative; } .hide.svelte-1xp0cw7 { display: none; } .compact.svelte-1xp0cw7 > \*, .compact.svelte-1xp0cw7 .box

{ border-radius: = 0px; } .compact.svelte-1xp0cw7, .panel.svelte-1xp0cw7 { border-radius: var(--conta= iner-radius);

background: var(--background-fill-secondary); padding: var(--= size-2); } .unequal-height.svelte-1xp0cw7 { align-items:

flex-start; } .stretch.svelte-1xp0cw7 { align-items: stretch; } .stretch.svelte-1xp0cw7 > .column > \*, .stretch.svelte-

1xp0cw7 > .column > = .form > \* { flex-grow: 1; flex-shrink: 0; } div.svelte-1xp0cw7 > \*, div.svelte-1xp0cw7 > .form >

\* { flex: 1 1 0%; flex= -wrap: wrap; min-width: min(160px, 100%); } .grow-children.svelte-1xp0cw7 > .column { align-

self: stretch; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type:

text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-

CRfssLQd.css @charset "utf-8"; div.svelte-lag733 .math.inline { fill: var(--body-text-color); display: inl= ine-block;

vertical-align: middle; padding: var(--size-1-5) -var(--size-1);= color: var(--body-text-color); } div.svelte-

lag733 .math.inline svg { display: inline; margin-bottom: 0.22em= ; } div.svelte-lag733 { max-width:

100%; } .hide.svelte-lag733 { display: none; } div.svelte-vuh1yp { transition: 0.15s; } .pending.svelte-vuh1yp { opacity:

0.2; } .padding.svelte-vuh1yp { padding: var(--block-padding); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Example-2\_XpXLIx.css @charset "utf-

8"; .gallery.svelte-1ayixqk { padding: var(--size-1) var(--size-2); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/ApiDocs-MshfCJX-.css @charset "utf-8"; .wrap.svelte-

e1ha0f.svelte-e1ha0f { padding: var(--size-6); } .attention.svelte-e1ha0f.svelte-e1ha0f { font-weight: var(--weight-bold);

f= ont-size: var(--text-lg); } .attention.svelte-e1ha0f code.svelte-e1ha0f { border: none; background: non= e; color:

var(--color-accent); font-weight: var(--weight-bold); } button.svelte-e1ha0f.svelte-e1ha0f { position: absolute; top:

var(--size-5)= ; right: var(--size-6); width: var(--size-4); color: var(--body-text-color)= ; } button.svelte-e1ha0f.svelte-

e1ha0f:hover { color: var(--color-accent); } @media (min-width: 768px) { button.svelte-e1ha0f.svelte-e1ha0f { top:

var(--size-6); } } h2.svelte-1i1gjw2.svelte-1i1gjw2 { display: flex; color: var(--body-text-co= lor); font-weight:

var(--weight-semibold); gap: var(--size-4); } h2.svelte-1i1gjw2 img.svelte-1i1gjw2 { margin-right: var(--size-2); width:

= var(--size-4); display: inline-block; } .url.svelte-1i1gjw2.svelte-1i1gjw2 { color: var(--color-accent); font-weigh= t:

400; } button.svelte-1i1gjw2.svelte-1i1gjw2 { position: absolute; top: var(--size-= 5); right: var(--size-6); width:

var(--size-4); color: var(--body-text-colo= r); } button.svelte-1i1gjw2.svelte-1i1gjw2:hover { color: var(--color-

accent); } @media (min-width: 768px) { button.svelte-1i1gjw2.svelte-1i1gjw2 { top: var(--size-6); } h2.svelte-1i1gjw2

img.svelte-1i1gjw2 { width: var(--size-5); } } .counts.svelte-1i1gjw2.svelte-1i1gjw2 { margin-top: auto; margin-right:

var= (--size-8); margin-bottom: auto; margin-left: auto; color: var(--body-text-= color); font-weight: var(--weight-light);

display: flex; flex-direction: ro= w; gap: 0.5rem; } .loading-dot.svelte-1i1gjw2.svelte-1i1gjw2 { position: relative; left: -

999= 9px; width: 12px; height: 12px; border-radius: 6px; background-color: rgb(2= 53, 123, 0); color: rgb(253, 123, 0);

box-shadow: 9999px 0px 0px -1px; marg= in-right: 0.3rem; } .self-baseline.svelte-1i1gjw2.svelte-1i1gjw2 { align-self:

baseline; } .title.svelte-1i1gjw2.svelte-1i1gjw2 { display: flex; flex-direction: row; = gap: 0.5rem; } .btn-text.svelte-

1i1gjw2.svelte-1i1gjw2 { font-size: var(--text-lg); } .hr.svelte-1yt946s { border: 0px; height: 1px; background:

var(--color-acce= nt-soft); margin-bottom: 12px; } .code.svelte-1yt946s { font-family: var(--font-mono); display:

inline; } .highlight.svelte-1yt946s { background: var(--color-accent-soft); color: va= r(--color-accent); padding:

var(--size-1); } .desc.svelte-1yt946s { color: var(--body-text-color-subdued); font-size: va= r(--text-lg); margin-top:

var(--size-1); } h4.svelte-1yt946s { display: flex; align-items: center; margin-top: var(--s= ize-6); margin-bottom:

var(--size-3); color: var(--body-text-color); font-w= eight: var(--weight-bold); } .toggle-icon.svelte-1yt946s { display:

flex; align-items: center; margin-ri= ght: var(--size-2); border-radius: var(--radius-full); background: var(--co= lor-grey-

300); width: 12px; height: 4px; } .toggle-dot.svelte-1yt946s { border-radius: var(--radius-full); background:= var(--color-

grey-700); width: 6px; height: 6px; margin-right: auto; } code.svelte-hq8ezf pre.svelte-hq8ezf { overflow-x: auto; color:

var(--body-= text-color); font-family: var(--font-mono); tab-size: 2; } code.svelte-hq8ezf.svelte-hq8ezf { position:

relative; display: block; } .copy.svelte-hq8ezf.svelte-hq8ezf { position: absolute; top: 0px; right: 0p= x; margin-top: -

5px; margin-right: -5px; } h3.svelte-1xzad0z { color: var(--body-text-color); font-weight: var(--secti= on-header-text-

weight); font-size: var(--text-lg); } .post.svelte-1xzad0z { margin-right: var(--size-2); border: 1px solid var(-= -border-

color-accent); border-radius: var(--radius-sm); background: var(--c= olor-accent-soft); padding-right: var(--size-1);

padding-bottom: var(--size= -1); padding-left: var(--size-1); color: var(--color-accent); font-weight: = var(--weight-

semibold); } .analytics.svelte-1xzad0z { color: var(--body-text-color-subdued); margin-t= op: var(--size-

1); } .desc.svelte-1xzad0z { color: var(--body-text-color); font-size: var(--text= -lg); margin-top: var(--size-1); }

code.svelte-114qcyq pre.svelte-114qcyq { overflow-x: auto; color: var(--bod= y-text-color); font-family: var(--font-

mono); tab-size: 2; } .token.string.svelte-114qcyq.svelte-114qcyq { display: contents; color: var= (--color-accent-base); }

code.svelte-114qcyq.svelte-114qcyq { position: relative; display: block; } .copy.svelte-114qcyq.svelte-114qcyq

{ position: absolute; top: 0px; right: = 0px; margin-top: -5px; margin-right: -5px; } .container.svelte-114qcyq.svelte-

114qcyq { display: flex; flex-direction: c= olumn; gap: var(--spacing-xxl); margin-top: var(--size-3); margin-bottom: v=

ar(--size-3); } .desc.svelte-114qcyq.svelte-114qcyq { color: var(--body-text-color-subdued)= ; } .api-name.svelte-

114qcyq.svelte-114qcyq { color: var(--color-accent); } code.svelte-j71ub0 pre.svelte-j71ub0 { overflow-x: auto; color:

var(--body-= text-color); font-family: var(--font-mono); tab-size: 2; } .token.string.svelte-j71ub0.svelte-j71ub0 { display:

contents; color: var(-= -color-accent-base); } code.svelte-j71ub0.svelte-j71ub0 { position: relative; display:

block; } .copy.svelte-j71ub0.svelte-j71ub0 { position: absolute; top: 0px; right: 0p= x; margin-top: -5px; margin-right: -

5px; } .container.svelte-j71ub0.svelte-j71ub0 { display: flex; flex-direction: col= umn; gap: var(--spacing-xxl); margin-

top: var(--size-3); margin-bottom: var= (--size-3); } .api-name.svelte-j71ub0.svelte-j71ub0 { color: var(--color-

accent); } .hr.svelte-16h224k { border: 0px; height: 1px; background: var(--color-acce= nt-soft); } .code.svelte-16h224k

{ font-family: var(--font-mono); margin-right: 10px; } .highlight.svelte-16h224k { background: var(--color-accent-soft);

color: va= r(--color-accent); padding: var(--size-1); } .desc.svelte-16h224k { color: var(--body-text-color-subdued); font-

size: va= r(--text-lg); } h4.svelte-16h224k { display: flex; align-items: center; margin-top: var(--s= ize-6); margin-

bottom: var(--size-3); color: var(--body-text-color); font-w= eight: var(--weight-bold); } .toggle-icon.svelte-16h224k

{ display: flex; align-items: center; margin-ri= ght: var(--size-2); border-radius: var(--radius-full); background: var(--co=

lor-grey-300); width: 12px; height: 4px; } .toggle-dot.svelte-16h224k { border-radius: var(--radius-full); background:=

var(--color-grey-700); width: 6px; height: 6px; margin-left: auto; } .tool-analytics.svelte-97rvku.svelte-97rvku.svelte-

97rvku { font-size: 0.95= em; color: var(--body-text-color-subdued); } .transport-selection.svelte-97rvku.svelte-

97rvku.svelte-97rvku { margin-bot= tom: var(--size-4); } .snippets.svelte-97rvku.svelte-97rvku.svelte-97rvku { display:

flex; align-= items: center; margin-bottom: var(--size-4); } .snippets.svelte-97rvku > .svelte-97rvku + .svelte-97rvku

{ margin-left: va= r(--size-2); } .transport-label.svelte-97rvku.svelte-97rvku.svelte-97rvku { font-weight: 6= 00; color:

var(--body-text-color); margin-right: var(--size-2); } .file-upload-section.svelte-97rvku.svelte-97rvku.svelte-97rvku

{ margin-top= : var(--size-4); padding: var(--size-3); border: 1px solid var(--border-col= or-primary); border-radius:

var(--radius-md); background: var(--background-= fill-secondary); } .checkbox-label.svelte-97rvku.svelte-97rvku.svelte-

97rvku { display: flex; = align-items: center; font-weight: 600; color: var(--body-text-color); curso= r: pointer; margin-

bottom: var(--size-2); } .checkbox.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin-right: var(--s= ize-2); width:

var(--size-4); height: var(--size-4); cursor: pointer; accen= t-color: var(--color-accent); } .checkbox.svelte-97rvku.svelte-

97rvku.svelte-97rvku:checked { background-co= lor: var(--color-accent); border-color: var(--color-accent); } .file-

upload-explanation.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin= : 0px; color: var(--body-text-

color); } .snippet.svelte-97rvku.svelte-97rvku.svelte-97rvku { display: flex; align-i= tems: center; border: 1px solid

var(--border-color-primary); border-radius:= var(--radius-md); padding: var(--size-1) var(--size-1-5); color: var(--bod=

y-text-color); line-height: 1; user-select: none; } .current-lang.svelte-97rvku.svelte-97rvku.svelte-97rvku { border: 1px

solid= var(--body-text-color-subdued); color: var(--body-text-color); } .inactive-lang.svelte-97rvku.svelte-97rvku.svelte-

97rvku { cursor: pointer;= color: var(--body-text-color-subdued); } .inactive-lang.svelte-97rvku.svelte-97rvku.svelte-

97rvku:hover, .inactive-l= ang.svelte-97rvku.svelte-97rvku.svelte-97rvku:focus { box-shadow: var(--sha= dow-drop);

color: var(--body-text-color); } code.svelte-97rvku pre.svelte-97rvku.svelte-97rvku { overflow-x: auto; colo= r:

var(--body-text-color); font-family: var(--font-mono); tab-size: 2; } .copy.svelte-97rvku.svelte-97rvku.svelte-97rvku

{ position: absolute; top: = 0px; right: 0px; margin-top: 5px; margin-right: 5px; z-index: 10; } .mcp-url.svelte-

97rvku.svelte-97rvku.svelte-97rvku { padding: var(--size-2)= ; position: relative; } .mcp-url.svelte-97rvku label.svelte-

97rvku.svelte-97rvku { display: block; = margin-bottom: var(--size-2); font-weight: 600; color: var(--body-text-colo=

r); } .mcp-url.svelte-97rvku .textbox.svelte-97rvku.svelte-97rvku { display: flex= ; align-items: center; gap: var(--size-

2); border: 1px solid var(--border-c= olor-primary); border-radius: var(--radius-sm); padding: var(--size-2); bac=

kground: var(--background-fill-primary); } .mcp-url.svelte-97rvku input.svelte-97rvku.svelte-97rvku { flex: 1 1 0%; bo=

rder: none; background: none; color: var(--body-text-color); font-family: v= ar(--font-mono); font-size: var(--text-md);

width: 100%; } .mcp-url.svelte-97rvku input.svelte-97rvku.svelte-97rvku:focus { outline: n= one; } .status-

indicator.svelte-97rvku.svelte-97rvku.svelte-97rvku { display: inli= ne-block; margin-right: var(--size-1-5); position:

relative; top: -1px; fon= t-size: 0.8em; } .status-indicator.active.svelte-97rvku.svelte-97rvku.svelte-97rvku { color:=

rgb(76, 175, 80); animation: 1s ease 0s infinite normal none running svelt= e-97rvku-pulse; } @keyframes svelte-

97rvku-pulse {=20 0% { opacity: 1; } 50% { opacity: 0.6; } 100% { opacity: 1; } } .mcp-tools.svelte-97rvku.svelte-

97rvku.svelte-97rvku { margin-top: var(--si= ze-4); border: 1px solid var(--border-color-primary); border-radius: var(--=

radius-md); overflow: hidden; } .tool-selection.svelte-97rvku.svelte-97rvku.svelte-97rvku { display: flex; = justify-

content: space-between; align-items: center; margin-bottom: var(--s= ize-2); } .tool-selection-controls.svelte-

97rvku.svelte-97rvku.svelte-97rvku { displa= y: flex; gap: var(--size-2); } .select-all-btn.svelte-97rvku.svelte-

97rvku.svelte-97rvku, .select-none-btn= .svelte-97rvku.svelte-97rvku.svelte-97rvku { padding: var(--size-1) var(--s= ize-

2); border: 1px solid var(--border-color-primary); border-radius: var(-= -radius-sm); background: var(--background-fill-

primary); color: var(--body-= text-color); cursor: pointer; font-size: var(--text-sm); } .select-all-btn.svelte-97rvku.svelte-

97rvku.svelte-97rvku:hover, .select-no= ne-btn.svelte-97rvku.svelte-97rvku.svelte-97rvku:hover { background: var(--=

background-fill-secondary); } .tool-item.svelte-97rvku.svelte-97rvku.svelte-97rvku { border-bottom: 1px s= olid

var(--border-color-primary); } .tool-item.svelte-97rvku.svelte-97rvku.svelte-97rvku:last-child { border-bo= ttom:

none; } .tool-header-wrapper.svelte-97rvku.svelte-97rvku.svelte-97rvku { display: f= lex; align-items: center; } .tool-

checkbox.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin-left: var= (--size-3); margin-right: var(--size-2); width:

var(--size-4); height: var(= --size-4); cursor: pointer; accent-color: var(--color-accent); } .tool-checkbox.svelte-

97rvku.svelte-97rvku.svelte-97rvku:checked { backgrou= nd-color: var(--color-accent); border-color: var(--color-

accent); } .tool-header.svelte-97rvku.svelte-97rvku.svelte-97rvku { flex: 1 1 0%; disp= lay: flex; justify-content: space-

between; align-items: center; padding: va= r(--size-3); background: var(--background-fill-primary); border: none; curs=

or: pointer; text-align: left; } .tool-header.svelte-97rvku.svelte-97rvku.svelte-97rvku:hover { background: =

var(--background-fill-secondary); } .tool-name.svelte-97rvku.svelte-97rvku.svelte-97rvku { font-family: var(--f= ont-

mono); font-weight: 600; } .tool-arrow.svelte-97rvku.svelte-97rvku.svelte-97rvku { color: var(--body-t= ext-color-

subdued); } .tool-content.svelte-97rvku.svelte-97rvku.svelte-97rvku { padding: var(--si= ze-3); background:

var(--background-fill-secondary); } .tool-description.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin-bottom= :

var(--size-3); color: var(--body-text-color); } .parameter.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin-bottom:

var(-= -size-2); padding: var(--size-2); background: var(--background-fill-primary= ); border-radius: var(--radius-

sm); } .parameter.svelte-97rvku code.svelte-97rvku.svelte-97rvku { font-weight: 60= 0; color: var(--color-

accent); } .parameter-type.svelte-97rvku.svelte-97rvku.svelte-97rvku { color: var(--bo= dy-text-color-subdued); margin-

left: var(--size-1); } .parameter-description.svelte-97rvku.svelte-97rvku.svelte-97rvku { margin-t= op: var(--size-1);

color: var(--body-text-color); } a.svelte-97rvku.svelte-97rvku.svelte-97rvku { text-decoration: underline; } .banner-

wrap.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { position: relat= ive; border-bottom: 1px solid var(--border-color-

primary); padding: var(--s= ize-4) var(--size-6); font-size: var(--text-md); } @media (min-width: 768px) { .banner-

wrap.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { font-size: va= r(--text-xl); } } .docs-wrap.svelte-1b8jye8.svelte-

1b8jye8.svelte-1b8jye8 { display: flex; fl= ex-direction: column; gap: var(--spacing-xxl); } .endpoint.svelte-

1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { border-radius: var= (--radius-md); background: var(--background-fill-primary);

padding-right: ;= padding-bottom: ; padding-left: ; padding-top: var(--size-1); font-size: v= ar(--text-md); } .client-

doc.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { padding-top: var= (--size-6); padding-right: var(--size-6); padding-

left: var(--size-6); font= -size: var(--text-md); } .snippets.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { display: flex;

ali= gn-items: center; margin-bottom: var(--size-4); } .snippets.svelte-1b8jye8 > .svelte-1b8jye8 + .svelte-1b8jye8

{ margin-left:= var(--size-2); } .snippet.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { display: flex; alig= n-items:

center; border: 1px solid var(--border-color-primary); border-radi= us: var(--radius-md); padding: var(--size-1)

var(--size-1-5); color: var(--= body-text-color); line-height: 1; user-select: none; font-size: var(--text-= lg); } .current-

lang.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { border: 1px so= lid var(--body-text-color-subdued); color:

var(--body-text-color); } .inactive-lang.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { cursor: point= er; color:

var(--body-text-color-subdued); } .inactive-lang.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8:hover, .inactiv= e-

lang.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8:focus { box-shadow: var= (--shadow-drop); color: var(--body-text-

color); } .snippet.svelte-1b8jye8 img.svelte-1b8jye8.svelte-1b8jye8 { margin-right: v= ar(--size-1-5); width: var(--size-

4); height: var(--size-4); } .endpoint-container.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { margin-t= op: var(--size-

3); margin-bottom: var(--size-3); border: 1px solid var(--bl= ock-border-color); border-radius: var(--radius-xl); padding-

right: ; paddin= g-bottom: ; padding-left: ; padding-top: 0px; } a.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { text-

decoration: underline= ; } p.padded.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { padding: 15px 0px; = font-size:

var(--text-lg); } code.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { font-size: var(--text-m= d); } .loading-dot.svelte-

1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { position: relat= ive; left: -9999px; width: 10px; height: 10px; border-radius:

5px; backgrou= nd-color: rgb(253, 123, 0); color: rgb(253, 123, 0); box-shadow: 9999px 0px= 0px -1px; margin-right:

0.25rem; } .docs-wrap .sm.secondary { padding-top: 1px; padding-bottom: 1px; } .self-baseline.svelte-1b8jye8.svelte-

1b8jye8.svelte-1b8jye8 { align-self: b= aseline; } .api-count.svelte-1b8jye8.svelte-1b8jye8.svelte-1b8jye8 { font-weight:

700;= color: rgb(253, 123, 0); align-self: baseline; font-family: var(--font-mon= o); font-size: var(--text-md); } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-VLJFaMM1.css @charset

"utf-8"; button.svelte-o34uqh, a.svelte-o34uqh { display: inline-flex; justify-conte= nt: center; align-items: center;

transition: var(--button-transition); padd= ing: var(--size-0-5) var(--size-2); text-align: center; } button.svelte-

o34uqh:hover { transform: var(--button-transform-hover); } button.svelte-o34uqh:active, a.svelte-o34uqh:active

{ transform: var(--butt= on-transform-active); } button[disabled].svelte-o34uqh, a.disabled.svelte-o34uqh { opacity: 0.5;

fi= lter: grayscale(30%); cursor: not-allowed; transform: none; } .hidden.svelte-o34uqh { display:

none; } .primary.svelte-o34uqh { border: var(--button-border-width) solid var(--but= ton-primary-border-color);

background: var(--button-primary-background-fill= ); color: var(--button-primary-text-color); box-shadow: var(--button-

primar= y-shadow); } .primary.svelte-o34uqh:hover, .primary[disabled].svelte-o34uqh { background= : var(--button-

primary-background-fill-hover); color: var(--button-primary-= text-color-hover); } .primary.svelte-o34uqh:hover

{ border-color: var(--button-primary-border-co= lor-hover); box-shadow: var(--button-primary-shadow-

hover); } .primary.svelte-o34uqh:active { box-shadow: var(--button-primary-shadow-act=

ive); } .primary[disabled].svelte-o34uqh { border-color: var(--button-primary-borde= r-color); } .secondary.svelte-

o34uqh { border: var(--button-border-width) solid var(--b= utton-secondary-border-color); background: var(--button-

secondary-backgroun= d-fill); color: var(--button-secondary-text-color); box-shadow: var(--butto= n-secondary-

shadow); } .secondary.svelte-o34uqh:hover, .secondary[disabled].svelte-o34uqh { backgr= ound: var(--button-

secondary-background-fill-hover); color: var(--button-se= condary-text-color-hover); } .secondary.svelte-o34uqh:hover

{ border-color: var(--button-secondary-borde= r-color-hover); box-shadow: var(--button-secondary-shadow-

hover); } .secondary.svelte-o34uqh:active { box-shadow: var(--button-secondary-shadow= -

active); } .secondary[disabled].svelte-o34uqh { border-color: var(--button-secondary-b= order-color); } .stop.svelte-

o34uqh { background: var(--button-cancel-background-fill); col= or: var(--button-cancel-text-color); border:

var(--button-border-width) sol= id var(--button-cancel-border-color); box-shadow: var(--button-cancel-shado=

w); } .stop.svelte-o34uqh:hover, .stop[disabled].svelte-o34uqh { background: var(= --button-cancel-background-fill-

hover); } .stop.svelte-o34uqh:hover { border-color: var(--button-cancel-border-color-= hover); box-shadow:

var(--button-cancel-shadow-hover); } .stop.svelte-o34uqh:active { box-shadow: var(--button-cancel-shadow-

active)= ; } .stop[disabled].svelte-o34uqh { border-color: var(--button-cancel-border-co= lor); } .sm.svelte-o34uqh

{ border-radius: var(--button-small-radius); padding: var= (--button-small-padding); font-weight: var(--button-small-

text-weight); fon= t-size: var(--button-small-text-size); } .md.svelte-o34uqh { border-radius: var(--button-medium-

radius); padding: va= r(--button-medium-padding); font-weight: var(--button-medium-text-weight); = font-size:

var(--button-medium-text-size); } .lg.svelte-o34uqh { border-radius: var(--button-large-radius); padding: var= (--button-

large-padding); font-weight: var(--button-large-text-weight); fon= t-size: var(--button-large-text-size); } .button-icon

{ width: var(--text-xl); height: var(--text-xl); } .button-icon.right-padded { margin-right: var(--spacing-

md); } .huggingface.svelte-o34uqh { background: rgb(20, 28, 46); color: rgb(255, 2= 55, 255); } .huggingface.svelte-

o34uqh:hover { background: rgb(40, 48, 66); color: rgb(= 255, 255, 255); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/ImagePreview-D5bl4ZzZ.css @charset "utf-8"; .image-

container.svelte-zxsjoa.svelte-zxsjoa { height: 100%; position: rela= tive; min-width: var(--size-20); } .image-

container.svelte-zxsjoa button.svelte-zxsjoa { width: var(--size-ful= l); height: var(--size-full); border-radius:

var(--radius-lg); display: fle= x; align-items: center; justify-content: center; } .image-frame.svelte-zxsjoa img { width:

var(--size-full); height: var(--siz= e-full); object-fit: scale-down; } .selectable.svelte-zxsjoa.svelte-zxsjoa { cursor:

crosshair; } .fullscreen-controls svg { position: relative; top: 0px; } .image-container:fullscreen { background-color:

rgb(0, 0, 0); display: flex= ; justify-content: center; align-items: center; } .image-container:fullscreen img { max-width:

90vw; max-height: 90vh; object= -fit: scale-down; } .image-frame.svelte-zxsjoa.svelte-zxsjoa { width: auto; height:

100%; displ= ay: flex; align-items: center; justify-content: center; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/Image-Ds1uFAjO.css @charset "utf-8"; img.svelte-

1pijsyv { object-fit: cover; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI----

Content-Type: text/css Content-Transfer-Encoding: quoted-printable Content-Location:

https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@400;600&display=swap @charset "utf-8"; @font-

face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 400; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-LCoeQqfX1RYOo3qNa7lqDY.woff2")

format("woff2"); unico= de-range: U+460-52F, U+1C80-1C8A, U+20B4, U+2DE0-2DFF, U+A640-A69F, U+FE2E-=

FE2F; } @font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 400; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-LCoeQqfX1RYOo3qPK7lqDY.woff2")

format("woff2"); unico= de-range: U+301, U+400-45F, U+490-491, U+4B0-4B1, U+2116; } @font-face { font-family:

"Source Sans Pro"; font-style: normal; font-weigh= t: 400; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-LCoeQqfX1RYOo3qNK7lqDY.woff2")

format("woff2"); unico= de-range: U+1F00-1FFF; } @font-face { font-family: "Source Sans Pro"; font-style: normal;

font-weigh= t: 400; font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-

LCoeQqfX1RYOo3qO67lqDY.woff2") format("woff2"); unico= de-range: U+370-377, U+37A-37F, U+384-38A, U

+38C, U+38E-3A1, U+3A3-3FF; } @font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 400;

font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-

LCoeQqfX1RYOo3qN67lqDY.woff2") format("woff2"); unico= de-range: U+102-103, U+110-111, U+128-129, U+168-

169, U+1A0-1A1, U+1AF-1B0,= U+300-301, U+303-304, U+308-309, U+323, U+329, U+1EA0-1EF9, U+20AB; }

@font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 400; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-LCoeQqfX1RYOo3qNq7lqDY.woff2")

format("woff2"); unico= de-range: U+100-2BA, U+2BD-2C5, U+2C7-2CC, U+2CE-2D7, U+2DD-2FF, U+304, U+3=

08, U+329, U+1D00-1DBF, U+1E00-1E9F, U+1EF2-1EFF, U+2020, U+20A0-20AB, U+20= AD-20C0, U+2113, U

+2C60-2C7F, U+A720-A7FF; } @font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 400;

font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xK3dSBYKcSV-

LCoeQqfX1RYOo3qOK7l.woff2") format("woff2"); unicode-= range: U+0-FF, U+131, U+152-153, U+2BB-2BC, U

+2C6, U+2DA, U+2DC, U+304, U+3= 08, U+329, U+2000-206F, U+20AC, U+2122, U+2191, U+2193, U+2212, U

+2215, U+F= EFF, U+FFFD; } @font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 600;

font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-

LCoeQqfX1RYOo3i54rwmhduz8A.woff2") format("woff2"); u= nicode-range: U+460-52F, U+1C80-1C8A, U+20B4, U

+2DE0-2DFF, U+A640-A69F, U+F= E2E-FE2F; } @font-face { font-family: "Source Sans Pro"; font-style: normal;

font-weigh= t: 600; font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-

LCoeQqfX1RYOo3i54rwkxduz8A.woff2") format("woff2"); u= nicode-range: U+301, U+400-45F, U+490-491, U+4B0-

4B1, U+2116; } @font-face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 600; font-display:

swap; src: url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-

LCoeQqfX1RYOo3i54rwmxduz8A.woff2") format("woff2"); u= nicode-range: U+1F00-1FFF; } @font-face { font-

family: "Source Sans Pro"; font-style: normal; font-weigh= t: 600; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-LCoeQqfX1RYOo3i54rwlBduz8A.woff2")

format("woff2"); u= nicode-range: U+370-377, U+37A-37F, U+384-38A, U+38C, U+38E-3A1, U+3A3-3FF;= } @font-

face { font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 600; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-LCoeQqfX1RYOo3i54rwmBduz8A.woff2")

format("woff2"); u= nicode-range: U+102-103, U+110-111, U+128-129, U+168-169, U+1A0-1A1, U+1AF-= 1B0, U

+300-301, U+303-304, U+308-309, U+323, U+329, U+1EA0-1EF9, U+20AB; } @font-face { font-family: "Source

Sans Pro"; font-style: normal; font-weigh= t: 600; font-display: swap; src: url("https://fonts.gstatic.com/s/sourcesan=

spro/v23/6xKydSBYKcSV-LCoeQqfX1RYOo3i54rwmRduz8A.woff2") format("woff2"); u= nicode-range: U+100-

2BA, U+2BD-2C5, U+2C7-2CC, U+2CE-2D7, U+2DD-2FF, U+304,= U+308, U+329, U+1D00-1DBF, U+1E00-1E9F,

U+1EF2-1EFF, U+2020, U+20A0-20AB, = U+20AD-20C0, U+2113, U+2C60-2C7F, U+A720-A7FF; } @font-face

{ font-family: "Source Sans Pro"; font-style: normal; font-weigh= t: 600; font-display: swap; src:

url("https://fonts.gstatic.com/s/sourcesan= spro/v23/6xKydSBYKcSV-LCoeQqfX1RYOo3i54rwlxdu.woff2")

format("woff2"); unic= ode-range: U+0-FF, U+131, U+152-153, U+2BB-2BC, U+2C6, U+2DA, U+2DC, U+304,= U

+308, U+329, U+2000-206F, U+20AC, U+2122, U+2191, U+2193, U+2212, U+2215,= U+FEFF, U+FFFD; } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Blocks-D0rbzfnq.css @charset "utf-

8"; .wrap.svelte-czcr5b.svelte-czcr5b { display: flex; flex-grow: 1; flex-direc= tion: column; width: var(--size-full); font-

weight: var(--body-text-weight)= ; font-size: var(--body-text-size); } .contain.svelte-czcr5b.svelte-czcr5b { display: flex;

flex-direction: colum= n; } footer.svelte-czcr5b.svelte-czcr5b { display: flex; justify-content: center= ; margin-top:

var(--size-4); color: var(--body-text-color-subdued); } .divider.svelte-czcr5b.svelte-czcr5b { margin-left: var(--size-1);

margin-r= ight: var(--size-2); } .show-api.svelte-czcr5b.svelte-czcr5b, .settings.svelte-czcr5b.svelte-czcr5=

b, .record.svelte-czcr5b.svelte-czcr5b { display: flex; align-items: center= ; } .show-api.svelte-czcr5b img.svelte-czcr5b

{ margin-right: var(--size-1); ma= rgin-left: var(--size-2); width: var(--size-3); } .settings.svelte-czcr5b img.svelte-czcr5b

{ margin-right: var(--size-1); ma= rgin-left: var(--size-1); width: var(--size-4); } .record.svelte-czcr5b img.svelte-czcr5b

{ margin-right: var(--size-1); marg= in-left: var(--size-1); width: var(--size-3); } .built-with.svelte-czcr5b.svelte-czcr5b

{ display: flex; align-items: cente= r; } .built-with.svelte-czcr5b.svelte-czcr5b:hover, .settings.svelte-czcr5b.svel= te-

czcr5b:hover, .record.svelte-czcr5b.svelte-czcr5b:hover { color: var(--b= ody-text-color); } .built-with.svelte-czcr5b

img.svelte-czcr5b { margin-right: var(--size-1); = margin-left: var(--size-1); margin-bottom: 1px; width: var(--size-

4); } .api-docs.svelte-czcr5b.svelte-czcr5b { display: flex; position: fixed; top= : 0px; right: 0px; z-index: var(--layer-

top); background: rgba(0, 0, 0, 0.5= ); width: var(--size-screen); height: var(--size-screen-h); } .backdrop.svelte-

czcr5b.svelte-czcr5b { flex: 1 1 0%; backdrop-filter: blur= (4px); } .api-docs-wrap.svelte-czcr5b.svelte-czcr5b { box-

shadow: var(--shadow-drop-= lg); background: var(--background-fill-primary); overflow: hidden auto; } @media (min-

width: 768px) { .api-docs-wrap.svelte-czcr5b.svelte-czcr5b { border-top-left-radius: var(= --radius-lg); border-bottom-

left-radius: var(--radius-lg); width: 950px; } } @media (min-width: 1536px) { .api-docs-wrap.svelte-czcr5b.svelte-

czcr5b { width: 1150px; } } #api-recorder-container.svelte-czcr5b.svelte-czcr5b { position: fixed; left= : 10px; bottom:

10px; z-index: 1000; } .show-api.svelte-czcr5b.svelte-czcr5b { display: flex; align-items: center;= } @media (max-

width: 640px) { .show-api.svelte-czcr5b.svelte-czcr5b, .show-api-divider.svelte-czcr5b.sv= elte-czcr5b { display:

none; } } .show-api.svelte-czcr5b.svelte-czcr5b:hover { color: var(--body-text-color)= ; } .hidden.svelte-czcr5b.svelte-

czcr5b { display: none; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI----

Content-Type: text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/theme.css?

v=2fa39f6df94156f7ef9d84370dbb2657d672827c75f082ac995879c32290c5ca @charset "utf-8"; @font-face { font-

family: "IBM Plex Mono"; src: url("static/fonts/IBMPlexMo= no/IBMPlexMono-Regular.woff2") format("woff2"); font-

style: normal; } @font-face { font-family: "IBM Plex Mono"; src: url("static/fonts/IBMPlexMo= no/IBMPlexMono-

Bold.woff2") format("woff2"); font-weight: bold; font-style:= normal; } :root { --name: default; --primary-50: #fff7ed; --

primary-100: #ffedd5; --p= rimary-200: #ffddb3; --primary-300: #fdba74; --primary-400: #fb923c; --prim= ary-500:

#f97316; --primary-600: #ea580c; --primary-700: #c2410c; --primary= -800: #9a3412; --primary-900: #7c2d12; --

primary-950: #6c2e12; --secondary-= 50: #eff6ff; --secondary-100: #dbeafe; --secondary-200: #bfdbfe; --secondar= y-

300: #93c5fd; --secondary-400: #60a5fa; --secondary-500: #3b82f6; --secon= dary-600: #2563eb; --secondary-700:

#1d4ed8; --secondary-800: #1e40af; --se= condary-900: #1e3a8a; --secondary-950: #1d3660; --neutral-50: #fafafa; --ne=

utral-100: #f4f4f5; --neutral-200: #e4e4e7; --neutral-300: #d4d4d8; --neutr= al-400: #bbbbc2; --neutral-500: #71717a; --

neutral-600: #52525b; --neutral-= 700: #3f3f46; --neutral-800: #27272a; --neutral-900: #18181b; --neutral-950= :

#0f0f11; --spacing-xxs: 1px; --spacing-xs: 2px; --spacing-sm: 4px; --spac= ing-md: 6px; --spacing-lg: 8px; --spacing-xl:

10px; --spacing-xxl: 16px; --= radius-xxs: 1px; --radius-xs: 2px; --radius-sm: 4px; --radius-md: 6px; --ra= dius-lg:

8px; --radius-xl: 12px; --radius-xxl: 22px; --text-xxs: 9px; --tex= t-xs: 10px; --text-sm: 12px; --text-md: 14px; --text-lg:

16px; --text-xl: 2= 2px; --text-xxl: 26px; --font: 'Source Sans Pro', ui-sans-serif, system-ui,= sans-serif; --font-mono:

'IBM Plex Mono', ui-monospace, Consolas, monospac= e; --body-background-fill: var(--background-fill-primary); --

body-text-colo= r: var(--neutral-800); --body-text-size: var(--text-md); --body-text-weight= : 400; --embed-radius:

var(--radius-sm); --color-accent: var(--primary-500)= ; --color-accent-soft: var(--primary-50); --background-fill-primary:

white;= --background-fill-secondary: var(--neutral-50); --border-color-accent: var= (--primary-300); --border-color-

primary: var(--neutral-200); --link-text-co= lor: var(--secondary-600); --link-text-color-active: var(--secondary-600); = --

link-text-color-hover: var(--secondary-700); --link-text-color-visited: v= ar(--secondary-500); --body-text-color-subdued:

var(--neutral-400); --accor= dion-text-color: var(--body-text-color); --table-text-color: var(--body-tex= t-color); --

shadow-drop: rgba(0,0,0,0.05) 0px 1px 2px 0px; --shadow-drop-lg= : 0 1px 3px 0 rgb(0 0 0 / 0.1), 0 1px 2px -1px rgb(0

0 0 / 0.1); --shadow-i= nset: rgba(0,0,0,0.05) 0px 2px 4px 0px inset; --shadow-spread: 3px; --block= -background-fill:

var(--background-fill-primary); --block-border-color: var= (--border-color-primary); --block-border-width: 1px; --block-

info-text-colo= r: var(--body-text-color-subdued); --block-info-text-size: var(--text-sm); = --block-info-text-weight:

400; --block-label-background-fill: var(--backgro= und-fill-primary); --block-label-border-color: var(--border-color-

primary);= --block-label-border-width: 1px; --block-label-shadow: var(--block-shadow)= ; --block-label-text-color:

var(--neutral-500); --block-label-margin: 0; --= block-label-padding: var(--spacing-sm) var(--spacing-lg); --block-label-

rad= ius: calc(var(--radius-sm) - 1px) 0 calc(var(--radius-sm) - 1px) 0; --block= -label-right-radius: 0 calc(var(--radius-

sm) - 1px) 0 calc(var(--radius-sm)= - 1px); --block-label-text-size: var(--text-sm); --block-label-text-weight= : 400; --

block-padding: var(--spacing-xl) calc(var(--spacing-xl) + 2px); --= block-radius: var(--radius-sm); --block-shadow:

none; --block-title-backgro= und-fill: none; --block-title-border-color: none; --block-title-border-widt= h: 0px; --block-

title-text-color: var(--neutral-500); --block-title-padding= : 0; --block-title-radius: none; --block-title-text-size: var(--text-

md); -= -block-title-text-weight: 400; --container-radius: var(--radius-sm); --form= -gap-width: 0px; --layout-gap:

var(--spacing-xxl); --panel-background-fill:= var(--background-fill-secondary); --panel-border-color: var(--border-

color= -primary); --panel-border-width: 0; --section-header-text-size: var(--text-= md); --section-header-text-weight:

400; --border-color-accent-subdued: var(= --primary-200); --code-background-fill: var(--neutral-100); --chatbot-text-=

size: var(--text-lg); --checkbox-background-color: var(--background-fill-pr= imary); --checkbox-background-color-

focus: var(--checkbox-background-color)= ; --checkbox-background-color-hover: var(--checkbox-background-color); --

ch= eckbox-background-color-selected: var(--color-accent); --checkbox-border-co= lor: var(--neutral-300); --checkbox-

border-color-focus: var(--color-accent)= ; --checkbox-border-color-hover: var(--neutral-300); --checkbox-border-colo= r-

selected: var(--color-accent); --checkbox-border-radius: var(--radius-sm)= ; --checkbox-border-width: var(--input-

border-width); --checkbox-label-back= ground-fill: var(--background-fill-primary); --checkbox-label-background-fi= ll-

hover: var(--background-fill-secondary); --checkbox-label-background-fil= l-selected: var(--checkbox-label-background-

fill); --checkbox-label-border-= color: var(--border-color-primary); --checkbox-label-border-color-hover: va=

r(--checkbox-label-border-color); --checkbox-label-border-color-selected: v= ar(--checkbox-label-border-color); --

checkbox-label-border-width: var(--inp= ut-border-width); --checkbox-label-gap: var(--spacing-lg); --checkbox-label= -

padding: var(--spacing-md) calc(2 \* var(--spacing-md)); --checkbox-label-s= hadow: none; --checkbox-label-text-size:

var(--text-md); --checkbox-label-t= ext-weight: 400; --checkbox-check: url("data:image/svg+xml,%3csvg viewBox=

=3D'0 0 16 16' fill=3D'white' xmlns=3D'http://www.w3.org/2000/svg'%3e%3cpat= h d=3D'M12.207 4.793a1 1 0 010

1.414l-5 5a1 1 0 01-1.414 0l-2-2a1 1 0 011.4= 14-1.414L6.5 9.086l4.293-4.293a1 1 0 011.414 0z'/%3e%3c/svg%3e"); --

radio-c= ircle: url("data:image/svg+xml,%3csvg viewBox=3D'0 0 16 16' fill=3D'white' =

xmlns=3D'http://www.w3.org/2000/svg'%3e%3ccircle cx=3D'8' cy=3D'8' r=3D'3'/= %3e%3c/svg%3e"); --checkbox-

shadow: var(--input-shadow); --checkbox-label-t= ext-color: var(--body-text-color); --checkbox-label-text-color-

selected: va= r(--checkbox-label-text-color); --error-background-fill: #fef2f2; --error-b= order-color: #b91c1c; --error-

border-width: 1px; --error-text-color: #b91c1= c; --error-icon-color: #b91c1c; --input-background-fill: white; --input-

bac= kground-fill-focus: var(--input-background-fill); --input-background-fill-h= over: var(--input-background-fill); --

input-border-color: var(--border-colo= r-primary); --input-border-color-focus: var(--secondary-300); --input-borde= r-

color-hover: var(--input-border-color); --input-border-width: 1px; --inpu= t-padding: var(--spacing-xl); --input-

placeholder-color: var(--neutral-400)= ; --input-radius: var(--radius-sm); --input-shadow: none; --input-shadow-fo= cus:

0 0 0 var(--shadow-spread) var(--secondary-50), var(--shadow-inset); -= -input-text-size: var(--text-md); --input-text-

weight: 400; --loader-color:= var(--color-accent); --prose-text-size: var(--text-md); --prose-text-weigh= t: 400; --prose-

header-text-weight: 600; --slider-color: var(--color-accent= ); --stat-background-fill: linear-gradient(to right,

var(--primary-400), va= r(--primary-200)); --table-border-color: var(--neutral-300); --table-even-b= ackground-fill:

white; --table-odd-background-fill: var(--neutral-50); --ta= ble-radius: var(--radius-sm); --table-row-focus: var(--color-

accent-soft); = --button-border-width: 0px; --button-cancel-background-fill: #ef4444; --but= ton-cancel-background-fill-

hover: #dc2626; --button-cancel-border-color: va= r(--button-secondary-border-color); --button-cancel-border-color-

hover: var= (--button-secondary-border-color-hover); --button-cancel-text-color: white;= --button-cancel-text-color-

hover: white; --button-cancel-shadow: var(--but= ton-secondary-shadow); --button-cancel-shadow-hover: var(--button-

secondary= -shadow-hover); --button-cancel-shadow-active: var(--button-secondary-shado= w-active); --button-

transform-hover: none; --button-transform-active: none;= --button-transition: all 0.2s ease; --button-large-padding:

var(--spacing-= lg) calc(2 \* var(--spacing-lg)); --button-large-radius: var(--radius-md); -= -button-large-text-size:

var(--text-lg); --button-large-text-weight: 600; -= -button-primary-background-fill: var(--primary-500); --button-primary-

backg= round-fill-hover: var(--primary-600); --button-primary-border-color: var(--= primary-500); --button-primary-

border-color-hover: var(--primary-500); --bu= tton-primary-text-color: white; --button-primary-text-color-hover:

var(--bu= tton-primary-text-color); --button-primary-shadow: none; --button-primary-s= hadow-hover: var(--button-

primary-shadow); --button-primary-shadow-active: = var(--button-primary-shadow); --button-secondary-background-

fill: var(--neu= tral-200); --button-secondary-background-fill-hover: var(--neutral-300); --= button-secondary-border-

color: var(--neutral-200); --button-secondary-borde= r-color-hover: var(--neutral-200); --button-secondary-text-color:

black; --= button-secondary-text-color-hover: var(--button-secondary-text-color); --bu= tton-secondary-shadow:

var(--button-primary-shadow); --button-secondary-sha= dow-hover: var(--button-secondary-shadow); --button-

secondary-shadow-active= : var(--button-secondary-shadow); --button-small-padding: var(--spacing-sm)= calc(1.5 \*

var(--spacing-sm)); --button-small-radius: var(--radius-md); --= button-small-text-size: var(--text-sm); --button-small-

text-weight: 400; --= button-medium-padding: var(--spacing-md) calc(2 \* var(--spacing-md)); --but= ton-medium-

radius: var(--radius-md); --button-medium-text-size: var(--text-= md); --button-medium-text-weight: 600; } :root .dark

{ --body-background-fill: var(--background-fill-primary); --bod= y-text-color: var(--neutral-100); --color-accent-soft:

var(--neutral-700); = --background-fill-primary: var(--neutral-950); --background-fill-secondary:= var(--neutral-900); --

border-color-accent: var(--neutral-600); --border-co= lor-primary: var(--neutral-700); --link-text-color-active:

var(--secondary-= 500); --link-text-color: var(--secondary-500); --link-text-color-hover: var= (--secondary-400); --link-

text-color-visited: var(--secondary-600); --body-= text-color-subdued: var(--neutral-400); --accordion-text-color:

var(--body-= text-color); --table-text-color: var(--body-text-color); --shadow-spread: 1= px; --block-background-fill:

var(--neutral-800); --block-border-color: var(= --border-color-primary); --block\_border\_width: None; --block-info-text-

colo= r: var(--body-text-color-subdued); --block-label-background-fill: var(--bac= kground-fill-secondary); --block-

label-border-color: var(--border-color-pri= mary); --block\_label\_border\_width: None; --block-label-text-color: var(--ne=

utral-200); --block\_shadow: None; --block\_title\_background\_fill: None; --bl= ock\_title\_border\_color: None; --

block\_title\_border\_width: None; --block-tit= le-text-color: var(--neutral-200); --panel-background-fill: var(--backgroun=

d-fill-secondary); --panel-border-color: var(--border-color-primary); --pan= el\_border\_width: None; --border-color-

accent-subdued: var(--border-color-ac= cent); --code-background-fill: var(--neutral-800); --checkbox-background-co=

lor: var(--neutral-800); --checkbox-background-color-focus: var(--checkbox-= background-color); --checkbox-

background-color-hover: var(--checkbox-backgr= ound-color); --checkbox-background-color-selected: var(--color-

accent); --c= heckbox-border-color: var(--neutral-700); --checkbox-border-color-focus: va= r(--color-accent); --

checkbox-border-color-hover: var(--neutral-600); --che= ckbox-border-color-selected: var(--color-accent); --checkbox-

border-width: = var(--input-border-width); --checkbox-label-background-fill: var(--neutral-= 800); --checkbox-label-

background-fill-hover: var(--checkbox-label-backgrou= nd-fill); --checkbox-label-background-fill-selected:

var(--checkbox-label-b= ackground-fill); --checkbox-label-border-color: var(--border-color-primary)= ; --checkbox-

label-border-color-hover: var(--checkbox-label-border-color); = --checkbox-label-border-color-selected: var(--checkbox-

label-border-color);= --checkbox-label-border-width: var(--input-border-width); --checkbox-label= -text-color:

var(--body-text-color); --checkbox-label-text-color-selected: = var(--checkbox-label-text-color); --error-background-fill:

var(--neutral-90= 0); --error-border-color: #ef4444; --error\_border\_width: None; --error-text= -color: #fef2f2; --error-

icon-color: #ef4444; --input-background-fill: var(= --neutral-800); --input\_background\_fill\_focus: None; --input-

background-fil= l-hover: var(--input-background-fill); --input-border-color: var(--border-c= olor-primary); --input-

border-color-focus: var(--neutral-700); --input-bord= er-color-hover: var(--input-border-color); --input\_border\_width:

None; --in= put-placeholder-color: var(--neutral-500); --input\_shadow: None; --input-sh= adow-focus: 0 0 0

var(--shadow-spread) var(--neutral-700), var(--shadow-ins= et); --loader\_color: None; --slider\_color: None; --stat-

background-fill: li= near-gradient(to right, var(--primary-400), var(--primary-600)); --table-bo= rder-color: var(--neutral-

700); --table-even-background-fill: var(--neutral= -950); --table-odd-background-fill: var(--neutral-900); --table-row-

focus: = var(--color-accent-soft); --button\_border\_width: None; --button-cancel-back= ground-fill: #b91c1c; --button-

cancel-background-fill-hover: #991b1b; --but= ton-cancel-border-color: var(--button-secondary-border-color); --button-

can= cel-border-color-hover: var(--button-secondary-border-color-hover); --butto= n-cancel-text-color: white; --button-

cancel-text-color-hover: white; --butt= on-cancel-shadow: var(--button-secondary-shadow); --button-cancel-shadow-ho=

ver: var(--button-secondary-shadow-hover); --button-cancel-shadow-active: v= ar(--button-secondary-shadow-active); --

button-primary-background-fill: var= (--primary-600); --button-primary-background-fill-hover: var(--primary-700)= ; --

button-primary-border-color: var(--primary-600); --button-primary-borde= r-color-hover: var(--primary-500); --button-

primary-text-color: white; --bu= tton-primary-text-color-hover: var(--button-primary-text-color); --button\_p=

rimary\_shadow: None; --button-primary-shadow-hover: var(--button-primary-sh= adow); --button-primary-shadow-

active: var(--button-primary-shadow); --butt= on-secondary-background-fill: var(--neutral-600); --button-secondary-

backgr= ound-fill-hover: var(--neutral-700); --button-secondary-border-color: var(-= -neutral-600); --button-secondary-

border-color-hover: var(--neutral-500); -= -button-secondary-text-color: white; --button-secondary-text-color-hover: v=

ar(--button-secondary-text-color); --button\_secondary\_shadow: None; --butto= n-secondary-shadow-hover: var(--button-

secondary-shadow); --button-secondar= y-shadow-active: var(--button-secondary-shadow); --name: default; --primary= -

50: #fff7ed; --primary-100: #ffedd5; --primary-200: #ffddb3; --primary-300= : #fdba74; --primary-400: #fb923c; --

primary-500: #f97316; --primary-600: #= ea580c; --primary-700: #c2410c; --primary-800: #9a3412; --primary-900:

#7c2= d12; --primary-950: #6c2e12; --secondary-50: #eff6ff; --secondary-100: #dbe= afe; --secondary-200: #bfdbfe; --

secondary-300: #93c5fd; --secondary-400: #= 60a5fa; --secondary-500: #3b82f6; --secondary-600: #2563eb; --

secondary-700= : #1d4ed8; --secondary-800: #1e40af; --secondary-900: #1e3a8a; --secondary-= 950: #1d3660; --

neutral-50: #fafafa; --neutral-100: #f4f4f5; --neutral-200:= #e4e4e7; --neutral-300: #d4d4d8; --neutral-400: #bbbbc2; --

neutral-500: #7= 1717a; --neutral-600: #52525b; --neutral-700: #3f3f46; --neutral-800: #2727= 2a; --neutral-900:

#18181b; --neutral-950: #0f0f11; --spacing-xxs: 1px; --s= pacing-xs: 2px; --spacing-sm: 4px; --spacing-md: 6px; --

spacing-lg: 8px; --= spacing-xl: 10px; --spacing-xxl: 16px; --radius-xxs: 1px; --radius-xs: 2px;= --radius-sm: 4px; --

radius-md: 6px; --radius-lg: 8px; --radius-xl: 12px; -= -radius-xxl: 22px; --text-xxs: 9px; --text-xs: 10px; --text-sm:

12px; --tex= t-md: 14px; --text-lg: 16px; --text-xl: 22px; --text-xxl: 26px; --font: 'So= urce Sans Pro', ui-sans-serif,

system-ui, sans-serif; --font-mono: 'IBM Ple= x Mono', ui-monospace, Consolas, monospace; --body-text-size:

var(--text-md= ); --body-text-weight: 400; --embed-radius: var(--radius-sm); --color-accen= t: var(--primary-500); --

shadow-drop: rgba(0,0,0,0.05) 0px 1px 2px 0px; --s= hadow-drop-lg: 0 1px 3px 0 rgb(0 0 0 / 0.1), 0 1px 2px -1px rgb(0

0 0 / 0.1= ); --shadow-inset: rgba(0,0,0,0.05) 0px 2px 4px 0px inset; --block-border-w= idth: 1px; --block-info-text-size:

var(--text-sm); --block-info-text-weight= : 400; --block-label-border-width: 1px; --block-label-shadow: var(--block-s=

hadow); --block-label-margin: 0; --block-label-padding: var(--spacing-sm) v= ar(--spacing-lg); --block-label-radius:

calc(var(--radius-sm) - 1px) 0 calc= (var(--radius-sm) - 1px) 0; --block-label-right-radius: 0 calc(var(--radius= -sm) -

1px) 0 calc(var(--radius-sm) - 1px); --block-label-text-size: var(--= text-sm); --block-label-text-weight: 400; --block-

padding: var(--spacing-xl= ) calc(var(--spacing-xl) + 2px); --block-radius: var(--radius-sm); --block-= shadow: none; --

block-title-background-fill: none; --block-title-border-col= or: none; --block-title-border-width: 0px; --block-title-

padding: 0; --bloc= k-title-radius: none; --block-title-text-size: var(--text-md); --block-titl= e-text-weight: 400; --

container-radius: var(--radius-sm); --form-gap-width:= 0px; --layout-gap: var(--spacing-xxl); --panel-border-width: 0; --

section-= header-text-size: var(--text-md); --section-header-text-weight: 400; --chat= bot-text-size: var(--text-lg); --

checkbox-border-radius: var(--radius-sm); = --checkbox-label-gap: var(--spacing-lg); --checkbox-label-padding:

var(--sp= acing-md) calc(2 \* var(--spacing-md)); --checkbox-label-shadow: none; --che= ckbox-label-text-size:

var(--text-md); --checkbox-label-text-weight: 400; -= -checkbox-check: url("data:image/svg+xml,%3csvg

viewBox=3D'0 0 16 16' fill= =3D'white' xmlns=3D'http://www.w3.org/2000/svg'%3e%3cpath d=3D'M12.207 4.79= 3a1

1 0 010 1.414l-5 5a1 1 0 01-1.414 0l-2-2a1 1 0 011.414-1.414L6.5 9.086l= 4.293-4.293a1 1 0 011.414 0z'/%3e%3c/svg

%3e"); --radio-circle: url("data:im= age/svg+xml,%3csvg viewBox=3D'0 0 16 16' fill=3D'white'

xmlns=3D'http://www= .w3.org/2000/svg'%3e%3ccircle cx=3D'8' cy=3D'8' r=3D'3'/%3e%3c/svg%3e"); --= checkbox-

shadow: var(--input-shadow); --error-border-width: 1px; --input-ba= ckground-fill-focus: var(--input-background-fill); --

input-border-width: 1p= x; --input-padding: var(--spacing-xl); --input-radius: var(--radius-sm); --= input-shadow:

none; --input-text-size: var(--text-md); --input-text-weight:= 400; --loader-color: var(--color-accent); --prose-text-size:

var(--text-md= ); --prose-text-weight: 400; --prose-header-text-weight: 600; --slider-colo= r: var(--color-accent); --table-

radius: var(--radius-sm); --button-border-w= idth: 0px; --button-transform-hover: none; --button-transform-active:

none;= --button-transition: all 0.2s ease; --button-large-padding: var(--spacing-= lg) calc(2 \* var(--spacing-lg)); --button-

large-radius: var(--radius-md); -= -button-large-text-size: var(--text-lg); --button-large-text-weight: 600; -= -button-

primary-shadow: none; --button-secondary-shadow: var(--button-prima= ry-shadow); --button-small-padding:

var(--spacing-sm) calc(1.5 \* var(--spac= ing-sm)); --button-small-radius: var(--radius-md); --button-small-text-size= :

var(--text-sm); --button-small-text-weight: 400; --button-medium-padding:= var(--spacing-md) calc(2 \* var(--spacing-

md)); --button-medium-radius: var= (--radius-md); --button-medium-text-size: var(--text-md); --button-medium-t= ext-

weight: 600; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type:

text/css Content-Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/Index-BkyjjaNE.css

@charset "utf-8"; .error.svelte-y6l4b.svelte-y6l4b { position: relative; padding: var(--size-= 4); color: var(--body-text-

color); text-align: center; } .error.svelte-y6l4b > .svelte-y6l4b { margin-top: var(--size-4); } a.svelte-y6l4b.svelte-y6l4b

{ color: var(--link-text-color); } a.svelte-y6l4b.svelte-y6l4b:hover { color: var(--link-text-color-hover); te= xt-decoration:

underline; } a.svelte-y6l4b.svelte-y6l4b:visited { color: var(--link-text-color-visited)= ; } a.svelte-y6l4b.svelte-

y6l4b:active { color: var(--link-text-color-active); } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/StreamingBar-Dq7cCzkm.css @charset "utf-8";

svg.svelte-zyxd38.svelte-zyxd38 { width: var(--size-20); height: var(--size= -20); } svg.svelte-zyxd38 path.svelte-

zyxd38 { fill: var(--loader-color); } div.svelte-zyxd38.svelte-zyxd38 { z-index: var(--layer-2); } .margin.svelte-

zyxd38.svelte-zyxd38 { margin: var(--size-4); } .wrap.svelte-btia7y.svelte-btia7y { display: flex; flex-direction: column;

= justify-content: center; align-items: center; z-index: var(--layer-3); tran= sition: opacity 0.1s ease-in-out; border-radius:

var(--block-radius); backg= round: var(--block-background-fill); padding: 0 var(--size-6); overflow: hi= dden; pointer-

events: none; } .wrap.center.svelte-btia7y.svelte-btia7y { top: 0px; right: 0px; left: 0px;= } .wrap.default.svelte-

btia7y.svelte-btia7y { inset: 0px; } .hide.svelte-btia7y.svelte-btia7y { opacity: 0; pointer-events:

none; } .generating.svelte-btia7y.svelte-btia7y { animation: 1s cubic-bezier(0.4, 0= , 0.6, 1) 0s 1 normal none running

svelte-btia7y-pulseStart, 2s cubic-bezie= r(0.4, 0, 0.6, 1) 1s infinite normal none running svelte-btia7y-pulse; bord= er:

2px solid var(--color-accent); background: transparent; z-index: var(--= layer-1); pointer-events:

none; } .translucent.svelte-btia7y.svelte-btia7y { background: none; } @keyframes svelte-btia7y-pulseStart {=20 0%

{ opacity: 0; } 100% { opacity: 1; } } @keyframes svelte-btia7y-pulse {=20 0%, 100% { opacity: 1; } 50% { opacity:

0.5; } } .loading.svelte-btia7y.svelte-btia7y { z-index: var(--layer-2); color: var(= --body-text-color); } .eta-bar.svelte-

btia7y.svelte-btia7y { position: absolute; inset: 0px; tran= sform-origin: left center; opacity: 0.8; z-index: var(--layer-1);

transitio= n: 10ms; background: var(--background-fill-secondary); } .progress-bar-wrap.svelte-btia7y.svelte-btia7y

{ border: 1px solid var(--bo= rder-color-primary); background: var(--background-fill-primary); width: 55.= 5%; height:

var(--size-4); } .progress-bar.svelte-btia7y.svelte-btia7y { transform-origin: left center; = background-color: var(--loader-

color); width: var(--size-full); height: var= (--size-full); } .progress-level.svelte-btia7y.svelte-btia7y { display: flex; flex-

direction= : column; align-items: center; z-index: var(--layer-2); width: var(--size-f= ull); } .progress-level-inner.svelte-

btia7y.svelte-btia7y { margin: var(--size-2) a= uto; color: var(--body-text-color); font-size: var(--text-sm); font-family:=

var(--font-mono); } .meta-text.svelte-btia7y.svelte-btia7y { position: absolute; bottom: 0px; r= ight: 0px; z-index:

var(--layer-2); padding: var(--size-1) var(--size-2); f= ont-size: var(--text-sm); font-family: var(--font-mono); } .meta-

text-center.svelte-btia7y.svelte-btia7y { display: flex; position: ab= solute; top: 0px; right: 0px; justify-content: center;

align-items: center;= transform: translateY(var(--size-6)); z-index: var(--layer-2); padding: va= r(--size-1) var(--size-2);

font-size: var(--text-sm); font-family: var(--fo= nt-mono); text-align: center; } .error.svelte-btia7y.svelte-btia7y { box-

shadow: var(--shadow-drop); border= : solid 1px var(--error-border-color); border-radius: var(--radius-full); b=

ackground: var(--error-background-fill); padding-right: var(--size-4); padd= ing-left: var(--size-4); color: var(--error-text-

color); font-weight: var(-= -weight-semibold); font-size: var(--text-lg); line-height: var(--line-lg); = font-family:

var(--font); } .validation-error.svelte-btia7y.svelte-btia7y { pointer-events: auto; color= : var(--error-text-color); font-

weight: var(--weight-semibold); line-height= : var(--line-lg); font-family: var(--font); position: absolute; background:=

var(--error-background-fill); top: 0px; right: 0px; z-index: var(--layer-3= ); padding: var(--size-1) var(--size-2); font-size:

var(--text-md); text-al= ign: center; border-bottom-left-radius: var(--radius-sm); border-bottom: 1p= x solid var(--error-

border-color); border-left: 1px solid var(--error-borde= r-color); display: flex; justify-content: space-between; align-

items: cente= r; gap: var(--spacing-xl); } .minimal.svelte-btia7y.svelte-btia7y { pointer-events: none; } .minimal.svelte-

btia7y .progress-text.svelte-btia7y { background: var(--blo= ck-background-fill); } .border.svelte-btia7y.svelte-btia7y

{ border: 1px solid var(--border-color-= primary); } .clear-status.svelte-btia7y.svelte-btia7y { position: absolute; display:

fl= ex; top: var(--size-2); right: var(--size-2); justify-content: flex-end; ga= p: var(--spacing-sm); z-index: var(--layer-

1); } .toast-body.svelte-fee5uc { display: flex; position: relative; right: 0px; = left: 0px; align-items: center; margin: auto;

border-radius: var(--containe= r-radius); overflow: hidden; pointer-events: auto; } .toast-body.error.svelte-fee5uc

{ border: 1px solid var(--color-red-700); b= ackground: var(--color-red-50); } .dark .toast-body.error.svelte-fee5uc

{ border: 1px solid var(--color-red-5= 00); background-color: var(--color-grey-950); } .toast-body.warning.svelte-fee5uc

{ border: 1px solid var(--color-yellow-70= 0); background: var(--color-yellow-50); } .dark .toast-body.warning.svelte-

fee5uc { border: 1px solid var(--color-yel= low-500); background-color: var(--color-grey-950); } .toast-body.info.svelte-

fee5uc { border: 1px solid var(--color-grey-700); b= ackground: var(--color-grey-50); } .dark .toast-body.info.svelte-

fee5uc { border: 1px solid var(--color-grey-5= 00); background-color: var(--color-grey-950); } .toast-

body.success.svelte-fee5uc { border: 1px solid var(--color-green-700= ); background: var(--color-green-

50); } .dark .toast-body.success.svelte-fee5uc { border: 1px solid var(--color-gre= en-500); background-color:

var(--color-grey-950); } .toast-title.svelte-fee5uc { display: flex; align-items: center; font-weigh= t: var(--weight-bold);

font-size: var(--text-lg); line-height: var(--line-s= m); } .toast-title.error.svelte-fee5uc { color: var(--color-red-

700); } .dark .toast-title.error.svelte-fee5uc { color: var(--color-red-50); } .toast-title.warning.svelte-fee5uc { color:

var(--color-yellow-700); } .dark .toast-title.warning.svelte-fee5uc { color: var(--color-yellow-50); } .toast-

title.info.svelte-fee5uc { color: var(--color-grey-700); } .dark .toast-title.info.svelte-fee5uc { color: var(--color-grey-

50); } .toast-title.success.svelte-fee5uc { color: var(--color-green-700); } .dark .toast-title.success.svelte-fee5uc { color:

var(--color-green-50); } .toast-close.svelte-fee5uc { margin: 0 var(--size-3); border-radius: var(--= size-3); padding: 0px

var(--size-1-5); font-size: var(--size-5); line-heigh= t: var(--size-5); } .toast-close.error.svelte-fee5uc { color: var(--color-

red-700); } .dark .toast-close.error.svelte-fee5uc { color: var(--color-red-500); } .toast-close.warning.svelte-fee5uc

{ color: var(--color-yellow-700); } .dark .toast-close.warning.svelte-fee5uc { color: var(--color-yellow-500); = } .toast-

close.info.svelte-fee5uc { color: var(--color-grey-700); } .dark .toast-close.info.svelte-fee5uc { color: var(--color-grey-

500); } .toast-close.success.svelte-fee5uc { color: var(--color-green-700); } .dark .toast-close.success.svelte-fee5uc

{ color: var(--color-green-500); } .toast-text.svelte-fee5uc { font-size: var(--text-lg); overflow-wrap: break= -word;

word-break: break-word; } .toast-text.error.svelte-fee5uc { color: var(--color-red-700); } .dark .toast-text.error.svelte-

fee5uc { color: var(--color-red-50); } .toast-text.warning.svelte-fee5uc { color: var(--color-yellow-700); } .dark .toast-

text.warning.svelte-fee5uc { color: var(--color-yellow-50); } .toast-text.info.svelte-fee5uc { color: var(--color-grey-

700); } .dark .toast-text.info.svelte-fee5uc { color: var(--color-grey-50); } .toast-text.success.svelte-fee5uc { color:

var(--color-green-700); } .dark .toast-text.success.svelte-fee5uc { color: var(--color-green-50); } .toast-details.svelte-

fee5uc { margin: var(--size-3) var(--size-3) var(--si= ze-3) 0; width: 100%; } .toast-icon.svelte-fee5uc { display: flex;

position: relative; flex-shrink:= 0; justify-content: center; align-items: center; margin: var(--size-2); bo= rder-radius:

var(--radius-full); padding-top: ; padding-right: ; padding-bo= ttom: ; padding-left: calc(var(--size-1) - 1px); width:

35px; height: 35px;= } .toast-icon.error.svelte-fee5uc { color: var(--color-red-700); } .dark .toast-icon.error.svelte-fee5uc

{ color: var(--color-red-500); } .toast-icon.warning.svelte-fee5uc { color: var(--color-yellow-700); } .dark .toast-

icon.warning.svelte-fee5uc { color: var(--color-yellow-500); } .toast-icon.info.svelte-fee5uc { color: var(--color-grey-

700); } .dark .toast-icon.info.svelte-fee5uc { color: var(--color-grey-500); } .toast-icon.success.svelte-fee5uc { color:

var(--color-green-700); } .dark .toast-icon.success.svelte-fee5uc { color: var(--color-green-500); } @keyframes svelte-

fee5uc-countdown {=20 0% { transform: scaleX(1); } 100% { transform: scaleX(0); } } .timer.svelte-fee5uc { position:

absolute; bottom: 0px; left: 0px; transfor= m-origin: 0px 0px; animation: 10s linear 0s 1 normal forwards running svelt=

e-fee5uc-countdown; width: 100%; height: var(--size-1); } .timer.error.svelte-fee5uc { background: var(--color-red-

700); } .dark .timer.error.svelte-fee5uc { background: var(--color-red-500); } .timer.warning.svelte-fee5uc { background:

var(--color-yellow-700); } .dark .timer.warning.svelte-fee5uc { background: var(--color-yellow-

500); } .timer.info.svelte-fee5uc { background: var(--color-grey-700); } .dark .timer.info.svelte-fee5uc { background:

var(--color-grey-500); } .timer.success.svelte-fee5uc { background: var(--color-green-700); } .dark .timer.success.svelte-

fee5uc { background: var(--color-green-500); } .hidden.svelte-fee5uc { display: none; } .toast-text.svelte-fee5uc a { text-

decoration: underline; } .toast-wrap.svelte-1jktj9z { --toast-top: var(--size-4); display: flex; pos= ition: fixed; top:

calc(var(--toast-top) + var(--size-4)); right: var(--siz= e-4); flex-direction: column; align-items: end; gap: var(--size-2); z-

index= : var(--layer-top); width: calc(100% - var(--size-8)); } @media (min-width: 640px) { .toast-wrap.svelte-1jktj9z

{ width: calc(var(--size-96) + var(--size-10))= ; } } .streaming-bar.svelte-roz8lq { position: absolute; bottom: 0px; left:

0px; = right: 0px; height: 4px; background-color: var(--primary-600); animation: a= uto linear 0s 1 normal forwards

running svelte-roz8lq-countdown; z-index: 1= ; } @keyframes svelte-roz8lq-countdown {=20 0% { transform:

translate(0px); } 100% { transform: translate(-100%); } } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/FullscreenButton-BYduS5IX.css @charset "utf-

8"; .block.svelte-1svsvh2 { position: relative; margin: 0px; box-shadow: var(--= block-shadow); border-width:

var(--block-border-width); border-color: var(-= -block-border-color); border-radius: var(--block-radius); background:

var(-= -block-background-fill); width: 100%; line-height: var(--line-sm); } .block.fullscreen.svelte-1svsvh2 { border-

radius: 0px; } .auto-margin.svelte-1svsvh2 { margin-left: auto; margin-right: auto; } .block.border\_focus.svelte-1svsvh2

{ border-color: var(--color-accent); } .block.border\_contrast.svelte-1svsvh2 { border-color: var(--body-text-

color= ); } .padded.svelte-1svsvh2 { padding: var(--block-padding); } .hidden.svelte-1svsvh2 { display:

none; } .flex.svelte-1svsvh2 { display: flex; flex-direction: column; } .hide-container.svelte-1svsvh2:not(.fullscreen)

{ margin: 0px; box-shadow: = none; --block-border-width: 0; background: transparent; padding: 0px; overf= low:

visible; } .resize-handle.svelte-1svsvh2 { position: absolute; bottom: 0px; right: 0px= ; width: 10px; height: 10px; fill:

var(--block-border-color); cursor: nwse-= resize; } .fullscreen.svelte-1svsvh2 { position: fixed; top: 0px; left: 0px; width:

1= 00vw; height: 100vh; z-index: 1000; overflow: auto; } .animating.svelte-1svsvh2 { animation: 0.1s ease-out 0s 1

normal forwards r= unning svelte-1svsvh2-pop-out; } @keyframes svelte-1svsvh2-pop-out {=20 0% { position: fixed;

top: var(--start-top); left: var(--start-left); wid= th: var(--start-width); height: var(--start-height); z-index: 100; } 100%

{ position: fixed; top: 0vh; left: 0vw; width: 100vw; height: 100vh;= z-index: 1000; } } .placeholder.svelte-1svsvh2

{ border-radius: var(--block-radius); border-wi= dth: var(--block-border-width); border-color: var(--block-border-color);

bo= rder-style: dashed; } div.svelte-j9uq24 > .md.prose { font-weight: var(--block-info-text-weight);= font-size:

var(--block-info-text-size); line-height: var(--line-sm); } div.svelte-j9uq24 > .md.prose \* { color: var(--block-info-text-

color); } div.svelte-j9uq24 { margin-bottom: var(--spacing-md); } span.has-info.svelte-g2oxp3 { margin-bottom:

var(--spacing-xs); } span.svelte-g2oxp3:not(.has-info) { margin-bottom: var(--spacing-lg); } span.svelte-g2oxp3

{ display: inline-block; position: relative; z-index: va= r(--layer-4); border: solid var(--block-title-border-width)

var(--block-tit= le-border-color); border-radius: var(--block-title-radius); background: var= (--block-title-background-

fill); padding: var(--block-title-padding); color= : var(--block-title-text-color); font-weight: var(--block-title-text-

weight= ); font-size: var(--block-title-text-size); line-height: var(--line-sm); } span[dir=3D"rtl"].svelte-g2oxp3 { display:

block; } .hide.svelte-g2oxp3 { margin: 0px; height: 0px; } label.svelte-j0zqjt.svelte-j0zqjt { display: inline-flex; align-

items: cent= er; z-index: var(--layer-2); box-shadow: var(--block-label-shadow); border-= right-color: ; border-right-

style: ; border-right-width: ; border-bottom-co= lor: ; border-bottom-style: ; border-bottom-width: ; border-image-

source: ;= border-image-slice: ; border-image-width: ; border-image-outset: ; border-= image-repeat: ; border-top: none;

border-left: none; border-radius: var(--b= lock-label-radius); background: var(--block-label-background-fill); padding= :

var(--block-label-padding); pointer-events: none; color: var(--block-labe= l-text-color); font-weight: var(--block-label-

text-weight); font-size: var(= --block-label-text-size); line-height: var(--line-sm); } .gr-group label.svelte-j0zqjt.svelte-

j0zqjt { border-top-left-radius: 0px; = } label.float.svelte-j0zqjt.svelte-j0zqjt { position: absolute; top: var(--bl= ock-

label-margin); left: var(--block-label-margin); } label.svelte-j0zqjt.svelte-j0zqjt:not(.float) { position: static; margin-to=

p: var(--block-label-margin); margin-left: var(--block-label-margin); } .hide.svelte-j0zqjt.svelte-j0zqjt { display: none; }

span.svelte-j0zqjt.svelte-j0zqjt { opacity: 0.8; margin-right: var(--size-2= ); width: calc(var(--block-label-text-size) -

1px); height: calc(var(--bloc= k-label-text-size) - 1px); } .hide-label.svelte-j0zqjt.svelte-j0zqjt { box-shadow: none;

border-width: 0= px; background: transparent; overflow: visible; } label[dir=3D"rtl"].svelte-j0zqjt.svelte-j0zqjt { border-

bottom-color: ; bor= der-bottom-style: ; border-bottom-width: ; border-left-color: ; border-left= -style: ; border-left-

width: ; border-image-source: ; border-image-slice: ;= border-image-width: ; border-image-outset: ; border-image-repeat:

; border= -top: none; border-right: none; border-bottom-left-radius: var(--block-radi= us); border-bottom-right-radius:

var(--block-label-radius); border-top-left= -radius: var(--block-label-radius); } label[dir=3D"rtl"].svelte-j0zqjt

span.svelte-j0zqjt { margin-left: var(--si= ze-2); margin-right: 0px; } .unstyled-link.svelte-1s8vnbx { color-scheme:

unset; forced-color-adjust: u= nset; mask: unset; math-depth: unset; position: unset; position-anchor: uns= et; text-size-

adjust: unset; appearance: unset; color: unset; font: unset; = font-palette: unset; font-synthesis: unset; position-area:

unset; text-orie= ntation: unset; text-rendering: unset; text-spacing-trim: unset; -webkit-fo= nt-smoothing: unset; -

webkit-locale: unset; -webkit-text-orientation: unset= ; -webkit-writing-mode: unset; writing-mode: unset; zoom: unset;

accent-col= or: unset; place-content: unset; place-items: unset; place-self: unset; ali= gnment-baseline: unset; anchor-

name: unset; anchor-scope: unset; animation-= composition: unset; animation: unset; app-region: unset; aspect-ratio:

unse= t; backdrop-filter: unset; backface-visibility: unset; background: unset; b= ackground-blend-mode: unset; baseline-

shift: unset; baseline-source: unset;= block-size: unset; border-block: unset; border: unset; border-radius: unse= t; border-

collapse: unset; border-end-end-radius: unset; border-end-start-r= adius: unset; border-inline: unset; border-start-end-

radius: unset; border-= start-start-radius: unset; inset: unset; box-decoration-break: unset; box-s= hadow: unset; box-

sizing: unset; break-after: unset; break-before: unset; b= reak-inside: unset; buffered-rendering: unset; caption-side:

unset; caret-a= nimation: unset; caret-color: unset; clear: unset; clip: unset; clip-path: = unset; clip-rule: unset; color-

interpolation: unset; color-interpolation-fi= lters: unset; color-rendering: unset; columns: unset; column-fill: unset; g= ap:

unset; column-rule: unset; column-span: unset; contain: unset; contain-= intrinsic-block-size: unset; contain-intrinsic-

size: unset; contain-intrins= ic-inline-size: unset; container: unset; content: unset; content-visibility= : unset; corner-

shape: unset; corner-block-end-shape: unset; corner-block-s= tart-shape: unset; counter-increment: unset; counter-reset:

unset; counter-= set: unset; cursor: pointer; cx: unset; cy: unset; d: unset; display: unset= ; dominant-baseline: unset;

dynamic-range-limit: unset; empty-cells: unset;= field-sizing: unset; fill: unset; fill-opacity: unset; fill-rule: unset; f=

ilter: unset; flex: unset; flex-flow: unset; float: unset; flood-color: uns= et; flood-opacity: unset; grid: unset; grid-area:

unset; height: unset; hyp= henate-character: unset; hyphenate-limit-chars: unset; hyphens: unset; imag= e-orientation:

unset; image-rendering: unset; initial-letter: unset; inline= -size: unset; inset-block: unset; inset-inline: unset; interest-

delay: unse= t; interpolate-size: unset; isolation: unset; letter-spacing: unset; lighti= ng-color: unset; line-break: unset;

list-style: unset; margin-block: unset;= margin: unset; margin-inline: unset; marker: unset; mask-type: unset; math= -

shift: unset; math-style: unset; max-block-size: unset; max-height: unset;= max-inline-size: unset; max-width: unset;

min-block-size: unset; min-heigh= t: unset; min-inline-size: unset; min-width: unset; mix-blend-mode: unset; = object-

fit: unset; object-position: unset; object-view-box: unset; offset: = unset; opacity: unset; order: unset; orphans: unset;

outline: unset; outlin= e-offset: unset; overflow-anchor: unset; overflow-block: unset; overflow-cl= ip-margin: unset;

overflow-inline: unset; overflow-wrap: unset; overflow: u= nset; overlay: unset; overscroll-behavior-block: unset;

overscroll-behavior= -inline: unset; overscroll-behavior: unset; padding-block: unset; padding: = unset; padding-inline:

unset; page: unset; page-orientation: unset; paint-o= rder: unset; perspective: unset; perspective-origin: unset; pointer-

events:= unset; position-try: unset; position-visibility: unset; print-color-adjust= : unset; quotes: unset; r: unset; reading-

flow: unset; reading-order: unset= ; resize: unset; rotate: unset; ruby-align: unset; ruby-position: unset; rx= : unset; ry:

unset; scale: unset; scroll-behavior: unset; scroll-initial-ta= rget: unset; scroll-margin-block: unset; scroll-margin: unset;

scroll-margi= n-inline: unset; scroll-marker-group: unset; scroll-padding-block: unset; s= croll-padding: unset; scroll-

padding-inline: unset; scroll-snap-align: unse= t; scroll-snap-stop: unset; scroll-snap-type: unset; scroll-target-group: u=

nset; scroll-timeline: unset; scrollbar-color: unset; scrollbar-gutter: uns= et; scrollbar-width: unset; shape-image-

threshold: unset; shape-margin: uns= et; shape-outside: unset; shape-rendering: unset; size: unset; speak: unset= ; stop-

color: unset; stop-opacity: unset; stroke: unset; stroke-dasharray: = unset; stroke-dashoffset: unset; stroke-linecap: unset;

stroke-linejoin: un= set; stroke-miterlimit: unset; stroke-opacity: unset; stroke-width: unset; = tab-size: unset; table-

layout: unset; text-align: unset; text-align-last: u= nset; text-anchor: unset; text-autospace: unset; text-box: unset; text-

comb= ine-upright: unset; text-decoration: unset; text-decoration-skip-ink: unset= ; text-emphasis: unset; text-emphasis-

position: unset; text-indent: unset; = text-overflow: unset; text-shadow: unset; text-transform: unset; text-under= line-

offset: unset; text-underline-position: unset; text-wrap: unset; timel= ine-scope: unset; touch-action: unset; transform:

unset; transform-box: uns= et; transform-origin: unset; transform-style: unset; transition: unset; tra= nslate: unset; user-

select: unset; vector-effect: unset; vertical-align: un= set; view-timeline: unset; view-transition-class: unset; view-

transition-gr= oup: unset; view-transition-name: unset; visibility: unset; border-spacing:= unset; -webkit-box-align:

unset; -webkit-box-decoration-break: unset; -web= kit-box-direction: unset; -webkit-box-flex: unset; -webkit-box-

ordinal-grou= p: unset; -webkit-box-orient: unset; -webkit-box-pack: unset; -webkit-box-r= eflect: unset; -webkit-line-

break: unset; -webkit-line-clamp: unset; -webki= t-mask-box-image: unset; -webkit-rtl-ordering: unset; -webkit-ruby-

position= : unset; -webkit-tap-highlight-color: unset; -webkit-text-combine: unset; -= webkit-text-decorations-in-effect:

unset; -webkit-text-fill-color: unset; -= webkit-text-security: unset; -webkit-text-stroke: unset; -webkit-user-drag:= unset;

white-space-collapse: unset; widows: unset; width: unset; will-chan= ge: unset; word-break: unset; word-spacing: unset;

x: unset; y: unset; z-in= dex: unset; } button.svelte-fuq25k { display: flex; justify-content: center; align-items:= center;

gap: 1px; z-index: var(--layer-2); border-radius: var(--radius-xs)= ; color: var(--block-label-text-color); border: 1px solid

var(--border-colo= r); padding: var(--spacing-xxs); } button.svelte-fuq25k:hover { background-color: var(--background-

fill-second= ary); } button[disabled].svelte-fuq25k { opacity: 0.5; box-shadow: none; } button[disabled].svelte-

fuq25k:hover { cursor: not-allowed; } .padded.svelte-fuq25k { background: var(--bg-color); } button.svelte-

fuq25k:hover, button.highlight.svelte-fuq25k { cursor: pointe= r; color: var(--color-accent); } .padded.svelte-

fuq25k:hover { color: var(--block-label-text-color); } span.svelte-fuq25k { padding: 0px 1px; font-size: 10px; }

div.svelte-fuq25k { display: flex; align-items: center; justify-content: ce= nter; transition: filter 0.2s ease-in-out; } .x-

small.svelte-fuq25k { width: 10px; height: 10px; } .small.svelte-fuq25k { width: 14px; height: 14px; } .medium.svelte-

fuq25k { width: 20px; height: 20px; } .large.svelte-fuq25k { width: 22px; height: 22px; } .pending.svelte-fuq25k

{ animation: 0.5s ease 0s infinite normal none runni= ng svelte-fuq25k-flash; } @keyframes svelte-fuq25k-flash {=20

0% { opacity: 0.5; } 50% { opacity: 1; } 100% { opacity: 0.5; } } .transparent.svelte-fuq25k { background: transparent;

border: none; box-sha= dow: none; } .empty.svelte-1oiin9d { display: flex; justify-content: center; align-items= : center;

margin-top: calc(0px - var(--size-6)); height: var(--size-full); = } .icon.svelte-1oiin9d { opacity: 0.5; height: var(--size-5);

color: var(--bo= dy-text-color); } .small.svelte-1oiin9d { min-height: calc(var(--size-32) - 20px); } .large.svelte-1oiin9d

{ min-height: calc(var(--size-64) - 20px); } .unpadded\_box.svelte-1oiin9d { margin-top: 0px; } .small\_parent.svelte-

1oiin9d { min-height: 100% !important; } .dropdown-arrow.svelte-xjn76a, .dropdown-arrow.svelte-1m886t3 { fill:

curre= ntcolor; } .circle.svelte-1m886t3 { fill: currentcolor; opacity: 0.1; } svg.svelte-184ngxt { animation: 1.5s linear 0s

infinite normal none running= svelte-184ngxt-spin; } @keyframes svelte-184ngxt-spin {=20 0% { transform:

rotate(0deg); } 100% { transform: rotate(360deg); } } h2.svelte-12ioyct { font-size: var(--text-xl) !important; } p.svelte-

12ioyct, h2.svelte-12ioyct { white-space: pre-line; } .wrap.svelte-12ioyct { display: flex; flex-direction: column; justify-

conte= nt: center; align-items: center; min-height: var(--size-60); color: var(--b= lock-label-text-color); line-height:

var(--line-md); height: 100%; padding-= top: var(--size-3); text-align: center; margin: auto var(--spacing-lg); } .or.svelte-

12ioyct { color: var(--body-text-color-subdued); display: flex; = } .icon-wrap.svelte-12ioyct { width: 30px; margin-

bottom: var(--spacing-lg); = } @media (min-width: 768px) { .wrap.svelte-12ioyct { font-size: var(--text-

lg); } } .hovered.svelte-12ioyct { color: var(--color-accent); } div.svelte-1ua59x3 { border-top: 1px solid transparent;

display: flex; max-= height: 100%; justify-content: center; gap: var(--spacing-sm); height: auto= ; align-items: flex-end;

color: var(--block-label-text-color); flex-shrink:= 0; } .show\_border.svelte-1ua59x3 { border-top: 1px solid var(--block-

border-colo= r); margin-top: var(--spacing-xxl); box-shadow: var(--shadow-drop); } .source-selection.svelte-snayfm

{ display: flex; align-items: center; justi= fy-content: center; border-top: 1px solid var(--border-color-primary); widt= h:

100%; margin-left: auto; margin-right: auto; height: var(--size-10); } .icon.svelte-snayfm { width: 22px; height: 22px;

margin: var(--spacing-lg) = var(--spacing-xs); padding: var(--spacing-xs); color: var(--neutral-400); b= order-radius:

var(--radius-md); } .selected.svelte-snayfm { color: var(--color-accent); } .icon.svelte-snayfm:hover, .icon.svelte-

snayfm:focus { color: var(--color-a= ccent); } .icon-button-wrapper.svelte-ud4hud { display: flex; flex-direction: row;

al= ign-items: center; justify-content: center; z-index: var(--layer-2); gap: v= ar(--spacing-sm); box-shadow:

var(--shadow-drop); border: 1px solid var(--b= order-color-primary); background: var(--block-background-fill); padding:

va= r(--spacing-xxs); } .icon-button-wrapper.hide-top-corner.svelte-ud4hud { border-top: none; bord= er-right: none;

border-radius: var(--block-label-right-radius); } .icon-button-wrapper.display-top-corner.svelte-ud4hud { border-radius:

var(= --radius-sm) 0 0 var(--radius-sm); top: var(--spacing-sm); right: -1px; } .icon-button-wrapper.svelte-

ud4hud:not(.top-panel) { border: 1px solid var(= --border-color-primary); border-radius: var(--radius-sm); } .top-

panel.svelte-ud4hud { position: absolute; top: var(--block-label-margi= n); right: var(--block-label-margin); margin:

0px; } .icon-button-wrapper.svelte-ud4hud button { margin: var(--spacing-xxs); bor= der-radius: var(--radius-xs);

position: relative; } .icon-button-wrapper.svelte-ud4hud a.download-link:not(:last-child), .icon-= button-wrapper.svelte-

ud4hud button:not(:last-child) { margin-right: var(--= spacing-xxs); } .icon-button-wrapper.svelte-ud4hud a.download-

link:not(:last-child):not(.no= -border \*)::after, .icon-button-wrapper.svelte-ud4hud button:not(:last-chil= d):not(.no-

border \*)::after { content: ""; position: absolute; right: -4.5p= x; top: 15%; height: 70%; width: 1px; background-color:

var(--border-color-= primary); } .icon-button-wrapper.svelte-ud4hud > \* { height: 100%; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/MarkdownCode-Ca9FjBJK.css @charset "utf-

8"; .gradio-container-5-50-0-dev0 .md code, .gradio-container-5-50-0-dev0 .md p= re { background: none; font-family:

var(--font-mono); font-size: var(--text= -sm); text-align: left; white-space: pre; word-spacing: normal; word-break:=

normal; overflow-wrap: normal; line-height: 1.5; tab-size: 2; hyphens: non= e; } .gradio-container-5-50-0-dev0 .md

pre[class\*=3D"language-"]::selection, .gr= adio-container-5-50-0-dev0 .md pre[class\*=3D"language-

"] ::selection, .grad= io-container-5-50-0-dev0 .md code[class\*=3D"language-"]::selection, .gradio= -container-5-50-0-

dev0 .md code[class\*=3D"language-"] ::selection { text-sh= adow: none; background: rgb(179, 212, 252); } .gradio-

container-5-50-0-dev0 .md pre { margin-right: 0px; margin-left: 0px= ; overflow: auto; position: relative; margin-top:

var(--spacing-sm); margin= -bottom: var(--spacing-sm); box-shadow: none; border: none; background: var= (--code-

background-fill); padding: var(--spacing-xxl); font-family: var(--f= ont-mono); text-shadow: none; border-radius:

var(--radius-sm); display: blo= ck; white-space: pre; } .gradio-container-5-50-0-dev0 .md :not(pre) > code { border-

radius: var(--r= adius-xs); white-space: normal; background: var(--code-background-fill); bo= rder: 1px solid

var(--panel-border-color); padding: var(--spacing-xxs) var(= --spacing-xs); } .gradio-container-5-50-0-

dev0 .md .token.comment, .gradio-container-5-50-0-= dev0 .md .token.prolog, .gradio-container-5-50-0-

dev0 .md .token.doctype, .= gradio-container-5-50-0-dev0 .md .token.cdata { color: rgb(112, 128, 144); = } .gradio-

container-5-50-0-dev0 .md .token.punctuation { color: rgb(153, 153,= 153); } .gradio-container-5-50-0-

dev0 .md .token.namespace { opacity: 0.7; } .gradio-container-5-50-0-dev0 .md .token.property, .gradio-container-5-50-

0= -dev0 .md .token.tag, .gradio-container-5-50-0-dev0 .md .token.boolean, .gr= adio-container-5-50-0-

dev0 .md .token.number, .gradio-container-5-50-0-dev0= .md .token.constant, .gradio-container-5-50-0-

dev0 .md .token.symbol, .gra= dio-container-5-50-0-dev0 .md .token.deleted { color: rgb(153, 0, 85); } .gradio-

container-5-50-0-dev0 .md .token.selector, .gradio-container-5-50-0= -dev0 .md .token.attr-name, .gradio-container-5-

50-0-dev0 .md .token.string= , .gradio-container-5-50-0-dev0 .md .token.char, .gradio-container-5-50-0-d=

ev0 .md .token.builtin, .gradio-container-5-50-0-dev0 .md .token.inserted {= color: rgb(102, 153, 0); } .gradio-container-

5-50-0-dev0 .md .token.atrule, .gradio-container-5-50-0-d= ev0 .md .token.attr-value, .gradio-container-5-50-0-

dev0 .md .token.keyword= { color: rgb(0, 119, 170); } .gradio-container-5-50-0-dev0 .md .token.function, .gradio-

container-5-50-0= -dev0 .md .token.class-name { color: rgb(221, 74, 104); } .gradio-container-5-50-0-

dev0 .md .token.regex, .gradio-container-5-50-0-de= v0 .md .token.important, .gradio-container-5-50-0-

dev0 .md .token.variable = { color: rgb(238, 153, 0); } .gradio-container-5-50-0-dev0 .md .token.important, .gradio-

container-5-50-= 0-dev0 .md .token.bold { font-weight: 700; } .gradio-container-5-50-0-dev0 .md .token.italic { font-

style: italic; } .gradio-container-5-50-0-dev0 .md .token.entity { cursor:

help; } .dark .md .token.comment, .dark .md .token.prolog, .dark .md .token.cdata {= color: rgb(92, 99,

112); } .dark .md .token.doctype, .dark .md .token.punctuation, .dark .md .token.en= tity { color: rgb(171, 178,

191); } .dark .md .token.attr-name, .dark .md .token.class-name, .dark .md .token.b=

oolean, .dark .md .token.constant, .dark .md .token.number, .dark .md .toke= n.atrule { color: rgb(209, 154,

102); } .dark .md .token.keyword { color: rgb(198, 120,

221); } .dark .md .token.property, .dark .md .token.tag, .dark .md .token.symbol, .=

dark .md .token.deleted, .dark .md .token.important { color: rgb(224, 108, =

117); } .dark .md .token.selector, .dark .md .token.string, .dark .md .token.char,

= .dark .md .token.builtin, .dark .md .token.inserted, .dark .md .token.regex= , .dark .md .token.attr-

value, .dark .md .token.attr-value > .token.punctua= tion { color: rgb(152, 195,

121); } .dark .md .token.variable, .dark .md .token.operator, .dark .md .token.func= tion { color: rgb(97, 175,

239); } .dark .md .token.url { color: rgb(86, 182, 194); } span.svelte-7ddecg div[class\*=3D"code\_wrap"] { position:

relative; } span.svelte-7ddecg span.katex { font-size: var(--text-lg); direction: ltr; = } span.svelte-7ddecg

div[class\*=3D"code\_wrap"] > button { z-index: 1; cursor:= pointer; border-bottom-left-radius: var(--radius-sm);

padding: var(--spaci= ng-md); width: 25px; height: 25px; position: absolute; right: 0px; } span.svelte-7ddecg .check

{ opacity: 0; z-index: var(--layer-top); transiti= on: opacity 0.2s; background: var(--code-background-fill); color:

var(--bod= y-text-color); position: absolute; top: var(--size-1-5); left: var(--size-1= -5); } span.svelte-7ddecg p:not(:first-

child) { margin-top: var(--spacing-xxl); } span.svelte-7ddecg .md-header-anchor { margin-left: -25px; padding-right: 8=

px; line-height: 1; color: var(--body-text-color-subdued); opacity: 0; } span.svelte-7ddecg h1:hover .md-header-anchor,

span.svelte-7ddecg h2:hover = .md-header-anchor, span.svelte-7ddecg h3:hover .md-header-anchor, span.svel= te-

7ddecg h4:hover .md-header-anchor, span.svelte-7ddecg h5:hover .md-heade= r-anchor, span.svelte-7ddecg

h6:hover .md-header-anchor { opacity: 1; } span.md.svelte-7ddecg .md-header-anchor > svg { color: var(--body-text-

colo= r-subdued); } span.svelte-7ddecg table { word-break: break-word; } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: http://localhost:7860/assets/init-DV\_66Kck.css @charset "utf-8"; .nav-holder.svelte-

cba45o.svelte-cba45o { padding: var(--size-2) 0; border-= bottom: solid 1px var(--border-color-primary); } nav.svelte-

cba45o.svelte-cba45o { display: flex; flex-wrap: wrap; gap: var(= --size-2); justify-content: flex-end; margin: 0px auto;

padding: 0 var(--si= ze-8); } nav.svelte-cba45o a.svelte-cba45o { padding: var(--size-1) var(--size-2); b= order-radius:

var(--block-radius); border-width: var(--block-border-width);= border-color: transparent; color: var(--body-text-color-

subdued); } nav.svelte-cba45o a.active.svelte-cba45o { color: var(--body-text-color); b= order-color: var(--block-border-

color); background-color: var(--block-backg= round-fill); } .gradio-container-5-50-0-dev0 { display: flex; position:

relative; flex-dir= ection: column; padding: 0px; min-height: 1px; overflow: hidden; color: var= (--button-secondary-

text-color); } .embed-container.svelte-cba45o.svelte-cba45o { margin: var(--size-4) 0px; b= order: 1px solid

var(--button-secondary-border-color); border-radius: var(-= -embed-radius); } .with-info.svelte-cba45o.svelte-cba45o

{ padding-bottom: var(--size-7); } .embed-container.svelte-cba45o > main.svelte-cba45o { padding: var(--size-4= ); }

main.svelte-cba45o.svelte-cba45o { display: flex; flex-grow: 1; flex-direct= ion: column; } .app.svelte-cba45o.svelte-

cba45o { position: relative; margin: auto; paddin= g: var(--size-4) var(--size-8); width: 100%; height: 100%; } @media

(min-width: 640px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 640px= ; } } @media (min-

width: 768px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 768px= ; } } @media (min-width:

1024px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 1024p= x; } } @media (min-width:

1280px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 1280p= x; } } @media (min-width:

1536px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 1536p= x; } } @media (min-width:

1920px) { .fillable.svelte-cba45o.svelte-cba45o:not(.fill\_width) { max-width: 1920p= x; } } .info.svelte-cba45o.svelte-

cba45o { display: flex; position: absolute; bott= om: 0px; justify-content: flex-start; border-top: 1px solid var(--button-

se= condary-border-color); padding: var(--size-1) var(--size-5); width: 100%; c= olor: var(--body-text-color-subdued);

font-size: var(--text-md); white-spac= e: nowrap; } .info.svelte-cba45o > span.svelte-cba45o { overflow-wrap: break-

word; displ= ay: block; word-break: keep-all; } .info.svelte-cba45o > span.svelte-cba45o:nth-child(1) { margin-right:

4px; = min-width: 0px; max-width: max-content; overflow: hidden; color: var(--body= -text-color); text-overflow:

ellipsis; white-space: nowrap; } .info.svelte-cba45o > span.svelte-cba45o:nth-child(2) { margin-right: 3px;

= } .info.svelte-cba45o > span.svelte-cba45o:nth-child(2), .info.svelte-cba45o = > span.svelte-cba45o:nth-child(3)

{ width: max-content; } .info.svelte-cba45o > span.svelte-cba45o:nth-child(3) { place-self: flex-en= d; margin-left: auto;

text-align: right; } .info.svelte-cba45o > span.svelte-cba45o:nth-child(1) { flex-shrink: 9; } .hidden-title.svelte-

cba45o.svelte-cba45o { position: absolute; left: var(-= -size-5); opacity: 0; background: var(--button-secondary-

background-fill); = padding-right: 4px; } .info.svelte-cba45o a.svelte-cba45o { color: var(--body-text-

color); } .title.svelte-cba45o.svelte-cba45o { font-size: var(--text-sm); font-family= : var(--font-mono); } .hf.svelte-

cba45o.svelte-cba45o { margin-left: 5px; } .space-logo.svelte-cba45o img.svelte-cba45o { display: inline-block;

margin= -bottom: 4px; height: 12px; } main.svelte-cba45o a.svelte-cba45o:hover { text-decoration: underline; } ------

MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-

Transfer-Encoding: quoted-printable Content-Location: http://localhost:7860/assets/index-CCA5X64I.css @charset "utf-

8"; .gradio-container-5-50-0-dev0, .gradio-container-5-50-0-dev0 \*, .gradio-con= tainer-5-50-0-dev0 ::before, .gradio-

container-5-50-0-dev0 ::after { box-si= zing: border-box; border-width: 0px; border-style: solid; } .gradio-container-5-

50-0-dev0 html { text-size-adjust: 100%; line-height: 1= .5; font-family: var(--font-sans); tab-size: 2; } body { margin:

0px; line-height: inherit; } .gradio-container-5-50-0-dev0 hr { border-top-width: 1px; height: 0px; colo= r:

inherit; } .gradio-container-5-50-0-dev0 abbr:where([title]) { text-decoration: underl= ine dotted; } .gradio-container-5-

50-0-dev0 h1, .gradio-container-5-50-0-dev0 h2, .gradio= -container-5-50-0-dev0 h3, .gradio-container-5-50-0-dev0

h4, .gradio-contai= ner-5-50-0-dev0 h5, .gradio-container-5-50-0-dev0 h6 { font-weight: inherit= ; font-size:

inherit; } .gradio-container-5-50-0-dev0 a { color: inherit; text-decoration: inherit;= } .gradio-container-5-50-0-dev0

b, .gradio-container-5-50-0-dev0 strong { fon= t-weight: bolder; } .gradio-container-5-50-0-dev0 code, .gradio-

container-5-50-0-dev0 kbd, .gra= dio-container-5-50-0-dev0 samp, .gradio-container-5-50-0-dev0 pre { } .gradio-

container-5-50-0-dev0 small { font-size: 80%; } .gradio-container-5-50-0-dev0 sub, .gradio-container-5-50-0-dev0 sup

{ posi= tion: relative; vertical-align: baseline; font-size: 75%; line-height: 0; } .gradio-container-5-50-0-dev0 sub

{ bottom: -0.25em; } .gradio-container-5-50-0-dev0 sup { top: -0.5em; } .gradio-container-5-50-0-dev0 table { border-

color: inherit; text-indent: 0= px; } .gradio-container-5-50-0-dev0 button, .gradio-container-5-50-0-dev0 input, = .gradio-

container-5-50-0-dev0 optgroup, .gradio-container-5-50-0-dev0 selec= t, .gradio-container-5-50-0-dev0 textarea

{ margin: 0px; padding: 0px; colo= r: inherit; font-weight: inherit; font-size: 100%; line-height: inherit; fo= nt-family:

inherit; } .gradio-container-5-50-0-dev0 button, .gradio-container-5-50-0-dev0 select = { text-transform: none; } .gradio-

container-5-50-0-dev0 button, .gradio-container-5-50-0-dev0 [type= =3D"button"], .gradio-container-5-50-0-dev0

[type=3D"reset"], .gradio-conta= iner-5-50-0-dev0 [type=3D"submit"] { appearance: button; background-image: = none;

background-color: transparent; } .gradio-container-5-50-0-dev0 progress { vertical-align: baseline; } .gradio-container-5-

50-0-dev0 ::-webkit-inner-spin-button, .gradio-containe= r-5-50-0-dev0 ::-webkit-outer-spin-button { height:

auto; } .gradio-container-5-50-0-dev0 [type=3D"search"] { appearance: textfield; ou= tline-offset: -2px; } .gradio-

container-5-50-0-dev0 ::-webkit-search-decoration { appearance: non= e; } .gradio-container-5-50-0-dev0 ::-webkit-file-

upload-button { appearance: bu= tton; font: inherit; } .gradio-container-5-50-0-dev0 summary { display: list-

item; } .gradio-container-5-50-0-dev0 blockquote, .gradio-container-5-50-0-dev0 dl,= .gradio-container-5-50-0-dev0

dd, .gradio-container-5-50-0-dev0 h1, .gradi= o-container-5-50-0-dev0 h2, .gradio-container-5-50-0-dev0 h3, .gradio-

conta= iner-5-50-0-dev0 h4, .gradio-container-5-50-0-dev0 h5, .gradio-container-5-= 50-0-dev0 h6, .gradio-container-5-

50-0-dev0 hr, .gradio-container-5-50-0-de= v0 figure, .gradio-container-5-50-0-dev0 p, .gradio-container-5-50-0-dev0 p=

re { margin: 0px; } .gradio-container-5-50-0-dev0 fieldset { margin: 0px; padding: 0px; } .gradio-container-5-50-0-dev0

legend { padding: 0px; } .gradio-container-5-50-0-dev0 ol, .gradio-container-5-50-0-dev0 ul, .gradio= -container-5-50-0-

dev0 menu { margin: 0px; padding: 0px; } .gradio-container-5-50-0-dev0 textarea { resize: vertical; } .gradio-container-

5-50-0-dev0 input::placeholder, .gradio-container-5-50-0-= dev0 textarea::placeholder { opacity: 1; color: --color-

var(--color-grey-40= 0); } .gradio-container-5-50-0-dev0 button, .gradio-container-5-50-0-dev0 [role= =3D"button"]

{ cursor: pointer; } .gradio-container-5-50-0-dev0 :disabled { cursor: default; } .gradio-container-5-50-0-dev0

img, .gradio-container-5-50-0-dev0 svg, .grad= io-container-5-50-0-dev0 video, .gradio-container-5-50-0-dev0

canvas, .grad= io-container-5-50-0-dev0 audio, .gradio-container-5-50-0-dev0 iframe, .grad= io-container-5-50-0-dev0

embed, .gradio-container-5-50-0-dev0 object { disp= lay: block; vertical-align: middle; } .gradio-container-5-50-0-dev0

img, .gradio-container-5-50-0-dev0 video { ma= x-width: 100%; height: auto; margin: 0px; } .gradio-container-5-50-0-

dev0 [hidden] { display: none; } .gradio-container-5-50-0-dev0 [type=3D"text"], .gradio-container-5-50-0-dev= 0

[type=3D"email"], .gradio-container-5-50-0-dev0 [type=3D"url"], .gradio-c= ontainer-5-50-0-dev0

[type=3D"password"], .gradio-container-5-50-0-dev0 [ty= pe=3D"number"], .gradio-container-5-50-0-dev0

[type=3D"date"], .gradio-cont= ainer-5-50-0-dev0 [type=3D"datetime-local"], .gradio-container-5-50-0-dev0 =

[type=3D"month"], .gradio-container-5-50-0-dev0 [type=3D"search"], .gradio-= container-5-50-0-dev0

[type=3D"tel"], .gradio-container-5-50-0-dev0 [type= =3D"time"], .gradio-container-5-50-0-dev0

[type=3D"week"], .gradio-containe= r-5-50-0-dev0 [multiple], .gradio-container-5-50-0-dev0 textarea, .gradio-c=

ontainer-5-50-0-dev0 select { --tw-shadow: 0 0 #0000; appearance: none; bor= der-width: 1px; border-color: rgb(107,

114, 128); border-radius: 0px; backg= round-color: rgb(255, 255, 255); padding: 0.5rem 0.75rem; font-size: 1rem; =

line-height: 1.5rem; } .gradio-container-5-50-0-dev0 [type=3D"checkbox"], .gradio-container-5-50-0= -dev0

[type=3D"radio"] { display: inline-block; flex-shrink: 0; vertical-al= ign: middle; appearance: none; border-width: 1px;

background-origin: border= -box; padding: 0px; width: 1rem; height: 1rem; color: rgb(37, 99, 235); use= r-select:

none; } .gradio-container-5-50-0-dev0 [type=3D"checkbox"]:checked { background-imag= e: url("data:image/svg+xml,

%3csvg viewBox=3D'0 0 16 16' fill=3D'white' xmln= s=3D'http://www.w3.org/2000/svg'%3e%3cpath d=3D'M12.207

4.793a1 1 0 010 1.4= 14l-5 5a1 1 0 01-1.414 0l-2-2a1 1 0 011.414-1.414L6.5 9.086l4.293-4.293a1 1= 0 011.414 0z'/%3e

%3c/svg%3e"); } .gradio-container-5-50-0-dev0 [type=3D"radio"]:checked { background-image: = url("data:image/svg

+xml,%3csvg viewBox=3D'0 0 16 16' fill=3D'white' xmlns= =3D'http://www.w3.org/2000/svg'%3e%3ccircle cx=3D'8'

cy=3D'8' r=3D'3'/%3e%3= c/svg%3e"); } .gradio-container-5-50-0-dev0 select { background-image: url("data:image/sv=

g+xml,%3csvg xmlns=3D'http://www.w3.org/2000/svg' fill=3D'none' viewBox=3D'= 0 0 20 20'%3e%3cpath

stroke=3D'%236b7280' stroke-linecap=3D'round' stroke-l= inejoin=3D'round' stroke-width=3D'1.5' d=3D'M6 8l4 4 4-4'/

%3e%3c/svg%3e"); = background-position: right 0.5rem center; background-size: 1.5em 1.5em; bac= kground-repeat:

no-repeat; padding-right: 2.5rem; } .gradio-container-5-50-0-dev0 [type=3D"checkbox"]:checked, .gradio-containe= r-5-

50-0-dev0 [type=3D"radio"]:checked { background-position: center center= ; background-size: 100% 100%;

background-repeat: no-repeat; } .gradio-container-5-50-0-dev0 [type=3D"checkbox"]:checked:hover, .gradio-co=

ntainer-5-50-0-dev0 [type=3D"checkbox"]:checked:focus, .gradio-container-5-= 50-0-dev0

[type=3D"radio"]:checked:hover, .gradio-container-5-50-0-dev0 [ty= pe=3D"radio"]:checked:focus { border-color:

transparent; } .gradio-container-5-50-0-dev0 [type=3D"checkbox"]:focus-visible, .gradio-co= ntainer-5-50-0-dev0

[type=3D"radio"]:focus-visible { outline: none; } .gradio-container-5-50-0-dev0 gradio-lite { display: flex; } .gradio-

container-5-50-0-dev0 gradio-lite:not(:defined) { display: none; } .gradio-container-5-50-0-dev0 .scroll-hide { scrollbar-

width: none; } .gradio-container-5-50-0-dev0 .sr-only { clip: rect(0px, 0px, 0px, 0px); po= sition: absolute; margin: -1px;

border-width: 0px; padding: 0px; width: 1px= ; height: 1px; overflow: hidden; white-space: nowrap; } .gradio-container-

5-50-0-dev0 .scroll-hide::-webkit-scrollbar { display: no= ne; } .gradio-container-5-50-0-dev0 { text-size-adjust: 100%;

line-height: 1.5; f= ont-family: var(--font); tab-size: 4; } .gradio-container-5-50-0-dev0 .cropper-container { position:

relative; touc= h-action: none; font-size: 0px; line-height: 0; direction: ltr; user-select= : none; } .gradio-container-5-50-

0-dev0 .cropper-container img { display: block; widt= h: 100%; height: 100%; min-width: 0px !important; max-width:

none !importan= t; min-height: 0px !important; max-height: none !important; } .gradio-container-5-50-0-dev0 .cropper-

wrap-box, .gradio-container-5-50-0-d= ev0 .cropper-canvas, .gradio-container-5-50-0-dev0 .cropper-drag-box, .grad= io-

container-5-50-0-dev0 .cropper-crop-box, .gradio-container-5-50-0-dev0 .= cropper-modal { position: absolute; inset:

0px; } .gradio-container-5-50-0-dev0 .cropper-wrap-box, .gradio-container-5-50-0-d= ev0 .cropper-canvas { overflow:

hidden; } .gradio-container-5-50-0-dev0 .cropper-drag-box { opacity: 0; background-co= lor: rgb(255, 255,

255); } .gradio-container-5-50-0-dev0 .cropper-modal { opacity: 0.5; background-col= or: rgb(0, 0, 0); } .gradio-

container-5-50-0-dev0 .cropper-view-box { display: block; outline: = rgba(51, 153, 255, 0.75) solid 1px; width: 100%;

height: 100%; overflow: hi= dden; } .gradio-container-5-50-0-dev0 .cropper-dashed { display: block; position: a=

bsolute; opacity: 0.5; border: 0px dashed rgb(238, 238, 238); } .gradio-container-5-50-0-dev0 .cropper-dashed.dashed-h

{ top: calc(33.3333%= ); left: 0px; border-top-width: 1px; border-bottom-width: 1px; width: 100%;= height:

calc(33.3333%); } .gradio-container-5-50-0-dev0 .cropper-dashed.dashed-v { top: 0px; left: ca= lc(33.3333%); border-

right-width: 1px; border-left-width: 1px; width: calc(= 33.3333%); height: 100%; } .gradio-container-5-50-0-

dev0 .cropper-center { display: block; position: a= bsolute; top: 50%; left: 50%; opacity: 0.75; width: 0px; height:

0px; } .gradio-container-5-50-0-dev0 .cropper-center::before, .gradio-container-5-= 50-0-dev0 .cropper-center::after

{ display: block; position: absolute; back= ground-color: rgb(238, 238, 238); content: " "; } .gradio-container-5-50-0-

dev0 .cropper-center::before { top: 0px; left: -3p= x; width: 7px; height: 1px; } .gradio-container-5-50-0-dev0 .cropper-

center::after { top: -3px; left: 0px= ; width: 1px; height: 7px; } .gradio-container-5-50-0-dev0 .cropper-face, .gradio-

container-5-50-0-dev0 = .cropper-line, .gradio-container-5-50-0-dev0 .cropper-point { display: bloc= k; position:

absolute; opacity: 0.1; width: 100%; height: 100%; } .gradio-container-5-50-0-dev0 .cropper-face { top: 0px; left: 0px;

backgrou= nd-color: rgb(255, 255, 255); } .gradio-container-5-50-0-dev0 .cropper-line { background-color: rgb(51,

153= , 255); } .gradio-container-5-50-0-dev0 .cropper-line.line-e { top: 0px; right: -3px;= cursor: ew-resize; width:

5px; } .gradio-container-5-50-0-dev0 .cropper-line.line-n { top: -3px; left: 0px; = cursor: ns-resize; height:

5px; } .gradio-container-5-50-0-dev0 .cropper-line.line-w { top: 0px; left: -3px; = cursor: ew-resize; width:

5px; } .gradio-container-5-50-0-dev0 .cropper-line.line-s { bottom: -3px; left: 0p= x; cursor: ns-resize; height:

5px; } .gradio-container-5-50-0-dev0 .cropper-point { opacity: 0.75; background-co= lor: rgb(51, 153, 255); width: 5px;

height: 5px; } .gradio-container-5-50-0-dev0 .cropper-point.point-e { top: 50%; right: -3p= x; cursor: ew-resize; margin-

top: -3px; } .gradio-container-5-50-0-dev0 .cropper-point.point-n { top: -3px; left: 50%= ; cursor: ns-resize; margin-

left: -3px; } .gradio-container-5-50-0-dev0 .cropper-point.point-w { top: 50%; left: -3px= ; cursor: ew-resize; margin-

top: -3px; } .gradio-container-5-50-0-dev0 .cropper-point.point-s { bottom: -3px; left: = 50%; cursor: s-resize; margin-

left: -3px; } .gradio-container-5-50-0-dev0 .cropper-point.point-ne { top: -3px; right: -= 3px; cursor: nesw-

resize; } .gradio-container-5-50-0-dev0 .cropper-point.point-nw { top: -3px; left: -3= px; cursor: nwse-resize; } .gradio-

container-5-50-0-dev0 .cropper-point.point-sw { bottom: -3px; left:= -3px; cursor: nesw-resize; } .gradio-container-5-50-

0-dev0 .cropper-point.point-se { right: -3px; bottom= : -3px; opacity: 1; cursor: nwse-resize; width: 20px; height:

20px; } @media (min-width: 768px) { .gradio-container-5-50-0-dev0 .cropper-point.point-se { width: 15px; heig= ht:

15px; } } @media (min-width: 992px) { .gradio-container-5-50-0-dev0 .cropper-point.point-se { width: 10px; heig= ht:

10px; } } @media (min-width: 1200px) { .gradio-container-5-50-0-dev0 .cropper-point.point-se { opacity: 0.75; wi= dth:

5px; height: 5px; } } .gradio-container-5-50-0-dev0 .cropper-point.point-se::before { display: bl= ock; position: absolute;

right: -50%; bottom: -50%; opacity: 0; background-= color: rgb(51, 153, 255); width: 200%; height: 200%; content: "

"; } .gradio-container-5-50-0-dev0 .cropper-invisible { opacity: 0; } .gradio-container-5-50-0-dev0 .cropper-bg

{ background-image: url("data:ima=

ge/png;base64,iVBORw0KGgoAAAANSUhEUgAAABAAAAAQAQMAAAAlPW0iAAAAA3NCSVQICAjb4=

U/gAAAABlBMVEXMzMz////TjRV2AAAACXBIWXMAAArrAAAK6wGCiw1aAAAAHHRFWHRTb2Z0d2Fy=

ZQBBZG9iZSBGaXJld29ya3MgQ1M26LyyjAAAABFJREFUCJlj+M/AgBVhF/0PAH6/D/HkDxOGAAA=

AAElFTkSuQmCC"); } .gradio-container-5-50-0-dev0 .cropper-hide { display: block; position: abs= olute; width: 0px;

height: 0px; } .gradio-container-5-50-0-dev0 .cropper-hidden { display: none !important; } .gradio-container-5-50-0-

dev0 .cropper-move { cursor: move; } .gradio-container-5-50-0-dev0 .cropper-crop { cursor: crosshair; } .gradio-

container-5-50-0-dev0 .cropper-disabled .cropper-drag-box, .gradio-= container-5-50-0-dev0 .cropper-disabled .cropper-

face, .gradio-container-5-= 50-0-dev0 .cropper-disabled .cropper-line, .gradio-container-5-50-0-dev0 .c= ropper-

disabled .cropper-point { cursor: not-allowed; } :root { --scale-0: 1rem; --scale-1: 1.125rem; --scale-2: 1.25rem; --scale-

3= : 1.5rem; --scale-4: 1.875rem; --scale-5: 2.25rem; --scale-6: 3rem; --scale= -7: 3.75rem; --scale-8: 4.5rem; --scale-9:

6rem; --scale-10: 8rem; --scale-= 000: .75rem; --scale-00: .875rem; --scale-fluid-0: clamp(.875rem, .8rem + .= 25vw,

1rem); --scale-fluid-1: clamp(1rem, .925rem + .25vw, 1.125rem); --sca= le-fluid-2: clamp(1.125rem, 1.05rem + .25vw,

1.25rem); --scale-fluid-3: cla= mp(1.25rem, 1.1rem + .5vw, 1.5rem); --scale-fluid-4: clamp(1.5rem, 1.275rem= + .75vw,

1.875rem); --scale-fluid-5: clamp(1.875rem, 1.65rem + .75vw, 2.25= rem); --scale-fluid-6: clamp(2.25rem, 1.8rem +

1.5vw, 3rem); --scale-fluid-= 7: clamp(3rem, 2.55rem + 1.5vw, 3.75rem); --scale-fluid-8: clamp(3.75rem, 3= .3rem +

1.5vw, 4.5rem); --scale-fluid-9: clamp(4.5rem, 3.6rem + 3vw, 6rem);= --scale-fluid-10: clamp(6rem, 4.8rem + 4vw,

8rem); --scale-fluid-000: clam= p(.625rem, .55rem + .25vw, .75rem); --scale-fluid-00: clamp(.75rem, .675rem=

+ .25vw, .875rem); --font-sans: IBM Plex Sans, ui-sans-serif, system-ui, -= apple-system, BlinkMacSystemFont, "Segoe

UI", Roboto, "Helvetica Neue", Ari= al, "Noto Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe = UI

Symbol", "Noto Color Emoji"; --font-serif: Georgia, Cambria, "Times New = Roman", Times, serif; --font-mono: IBM

Plex Mono, ui-monospace, SFMono-Regu= lar, Menlo, Monaco, Consolas, "Liberation Mono", "Courier New",

monospace; = --weight-light: 300; --weight-regular: 400; --weight-medium: 500; --weight-= semibold: 600; --weight-

bold: 700; --weight-extrabold: 800; --weight-black:= 900; --line-none: 1; --line-xs: 1.125; --line-sm: 1.4; --line-md:

1.5; --l= ine-lg: 1.625; --line-xl: 2; --letter-xs: -.05em; --letter-sm: -.025em; --l= etter-none: 0em; --letter-lg: .025em; --

letter-xl: .05em; --prose-xs: 45ch;= --prose-sm: 55ch; --prose-md: 65ch; --prose-lg: 75ch; --prose-xl: 85ch; --= size-1:

4px; --size-2: 8px; --size-3: 12px; --size-4: 16px; --size-5: 20px;= --size-6: 24px; --size-7: 28px; --size-8: 32px; --size-9:

36px; --size-10:= 40px; --size-11: 44px; --size-12: 48px; --size-14: 56px; --size-16: 64px; = --size-20: 80px; --size-24:

96px; --size-28: 112px; --size-32: 128px; --siz= e-36: 144px; --size-40: 160px; --size-44: 176px; --size-48: 192px; --size-

5= 2: 208px; --size-56: 224px; --size-60: 240px; --size-64: 256px; --size-72: = 288px; --size-80: 320px; --size-96:

384px; --size-px: 1px; --size-full: 100= %; --size-screen: 100vw; --size-min: min-content; --size-max: max-content; = --

size-0-5: 2px; --size-1-5: 6px; --size-2-5: 10px; --size-screen-h: 100vh;= --width-xs: 480px; --width-sm: 640px; --width-

md: 768px; --width-lg: 1024p= x; --width-xl: 1280px; --ratio-square: 1/1; --ratio-portrait: 3/4; --ratio-= landscape: 4/3; --

ratio-tall: 2/3; --ratio-wide: 3/2; --ratio-widescreen: 1= 6/9; --ratio-golden: 1.618/1; --radius-100: 100%; --radius-xs:

2px; --radiu= s-sm: 4px; --radius-md: 6px; --radius-lg: 8px; --radius-xl: 12px; --radius-= full: 9999px; --radius-2xl:

16px; --radius-3xl: 22px; --blur-xs: blur(4px);= --blur-sm: blur(8px); --blur-md: blur(16px); --blur-lg: blur(24px); --

blur= -xl: blur(40px); --layer-1: 10; --layer-2: 20; --layer-3: 30; --layer-4: 40= ; --layer-5: 50; --layer-below: -1; --layer-

top: 2147483647; --shadow-xs: 0= 1px 3px 0 rgba(0, 0, 0, .1), 0 1px 2px 0 rgba(0, 0, 0, .06); --shadow-sm: = 0 4px 6px -

2px rgba(0, 0, 0, .1), 0 2px 4px -2px rgba(0, 0, 0, .06); --shad= ow-md: 0 12px 16px -4px rgba(0, 0, 0, .1), 0 4px 6px -

2px rgba(0, 0, 0, .05= ); --shadow-lg: 0 20px 24px -4px rgba(0, 0, 0, .1), 0 8px 8px -4px rgba(0, = 0, 0, .04); --shadow-

xl: 0 24px 48px -12px rgba(0, 0, 0, .25); --ease-in-si= ne: cubic-bezier(.47, 0, .745, .715); --ease-out-sine: cubic-

bezier(.39, .5= 75, .565, 1); --ease-in-out-sine: cubic-bezier(.445, .05, .55, .95); --ease= -in-quad: cubic-

bezier(.55, .085, .68, .53); --ease-out-quad: cubic-bezier(= .25, .46, .45, .94); --ease-in-out-quad: cubic-

bezier(.455, .03, .515, .955= ); --ease-in-cubic: cubic-bezier(.55, .055, .675, .19); --ease-out-cubic: c= ubic-

bezier(.215, .61, .355, 1); --ease-in-out-cubic: cubic-bezier(.645, .0= 45, .355, 1); --ease-in-quart: cubic-

bezier(.895, .03, .685, .22); --ease-o= ut-quart: cubic-bezier(.165, .84, .44, 1); --ease-in-out-quart: cubic-bezie= r(.77,

0, .175, 1); --ease-in-quint: cubic-bezier(.755, .05, .855, .06); --= ease-out-quint: cubic-bezier(.23, 1, .32, 1); --ease-in-

out-quint: cubic-be= zier(.86, 0, .07, 1); --ease-in-expo: cubic-bezier(.95, .05, .795, .035); -= -ease-out-expo: cubic-

bezier(.19, 1, .22, 1); --ease-in-out-expo: cubic-bez= ier(1, 0, 0, 1); --ease-in-circ: cubic-bezier(.6, .04, .98, .335); --ease-

o= ut-circ: cubic-bezier(.075, .82, .165, 1); --ease-in-out-circ: cubic-bezier= (.785, .135, .15, .86); --ease-in-back: cubic-

bezier(.6, -.28, .735, .045);= --ease-out-back: cubic-bezier(.175, .885, .32, 1.275); --ease-in-out-back:= cubic-

bezier(.68, -.55, .265, 1.55); --easing-standard: cubic-bezier(.4, 0= , .2, 1); --easing-accelerate: cubic-bezier(.4, 0, 1, 1); --

easing-decelera= te: cubic-bezier(0, 0, .2, 1); --elevation-1: 0 1px 2px 0 rgba(0, 0, 0, .05= ); --elevation-2: 0 1px 3px 0

rgba(0, 0, 0, .1), 0 1px 2px 0 rgba(0, 0, 0, = .06); --elevation-3: 0 4px 6px -2px rgba(0, 0, 0, .1), 0 2px 4px -2px rgba(= 0,

0, 0, .06); --elevation-4: 0 12px 16px -4px rgba(0, 0, 0, .1), 0 4px 6px= -2px rgba(0, 0, 0, .05); --elevation-5: 0 20px

24px -4px rgba(0, 0, 0, .1)= , 0 8px 8px -4px rgba(0, 0, 0, .04); --elevation-6: 0 24px 48px -12px rgba(= 0, 0, 0, .25); --

elevation-7: 0 32px 64px -12px rgba(0, 0, 0, .2); --color-= grey-50: #f9fafb; --color-grey-100: #f3f4f6; --color-grey-200:

#e5e7eb; --c= olor-grey-300: #d1d5db; --color-grey-400: #9ca3af; --color-grey-500: #6b728= 0; --color-grey-600:

#4b5563; --color-grey-700: #374151; --color-grey-800: = #1f2937; --color-grey-900: #111827; --color-black:

#14141b; --color-grey: #= 6b7280; --color-red-300: #fca5a5; --color-red-500: #ef4444; --color-red-700= : #b91c1c; --

color-red: #ef4444; --color-green-300: #86efac; --color-green-= 500: #22c55e; --color-green-700: #15803d; --color-

green: #22c55e; --color-b= lue-300: #93c5fd; --color-blue-500: #0ea5e9; --color-blue-700: #1d4ed8; --c= olor-blue:

#0ea5e9; --color-pink-300: #fbb6ce; --color-pink-500: #ed64a6; -= -color-pink-700: #d53f8c; --color-pink: var(--color-

pink-500); --color-purp= le-300: #b794f4; --color-purple-500: #805ad5; --color-purple-700: #6b46c1; = --color-purple:

var(--color-purple-500); --color-teal-300: #81e6d9; --color= -teal-500: #38b2ac; --color-teal-700: #2c7a7b; --color-teal:

var(--color-te= al-500); --color-yellow-300: #fde047; --color-yellow-500: #eab308; --color-= yellow-700: #a16207; --

color-yellow: #eab308; --color-orange-300: #ffb066; = --color-orange-500: #ff7c00; --color-orange-700: #ce6400; --

color-orange: #= f97316; --color-brown-300: #a1887f; --color-brown-500: #795548; --color-bro= wn-700: #5d4037; --

color-brown: var(--color-brown-500); --color-blue-10: #f= afcff; --color-blue-50: #eff6ff; --color-blue-100: #dbeafe; --

color-blue-20= 0: #bfdbfe; --color-blue-400: #60a5fa; --color-blue-600: #2563eb; --color-b= lue-800: #1e40af; --color-

blue-900: #1e3a8a; --color-blue-950: #1c366b; --c= olor-grey-10: #fdfdfe; --color-grey-950: #0b0f19; --color-red-10:

#fffbfb; = --color-red-50: #fef2f2; --color-red-100: #fee2e2; --color-red-200: #fecaca= ; --color-red-400: #f87171; --

color-red-600: #dc2626; --color-red-800: #991= b1b; --color-red-900: #7f1d1d; --color-red-950: #63171a; --color-green-

10: = #f9fefc; --color-green-50: #ecfdf5; --color-green-100: #d1fae5; --color-gre= en-200: #bbf7d0; --color-green-400:

#4ade80; --color-green-600: #16a34a; --= color-green-800: #166534; --color-green-900: #14532d; --color-green-950:

#1= 34227; --color-orange-10: #fffbf6; --color-orange-50: #fff2e5; --color-oran= ge-100: #ffe5cc; --color-orange-200:

#ffd8b4; --color-orange-400: #ff9633; = --color-orange-600: #ee7400; --color-orange-800: #a45000; --color-orange-90=

0: #5c2d00; --color-orange-950: #3c1f00; --color-yellow-10: #fffef8; --colo= r-yellow-50: #fffbeb; --color-yellow-100:

#fff9c2; --color-yellow-200: #fef= 08a; --color-yellow-400: #facc15; --color-yellow-600: #ca8a04; --color-yell= ow-800:

#854d0e; --color-yellow-900: #713f12; --color-yellow-950: #633112; = --grid-2: repeat(2, minmax(0, 1fr)); --grid-3:

repeat(3, minmax(0, 1fr)); -= -grid-4: repeat(4, minmax(0, 1fr)); --grid-5: repeat(5, minmax(0, 1fr)); --= grid-6: repeat(6,

minmax(0, 1fr)); --grid-7: repeat(7, minmax(0, 1fr)); --g= rid-8: repeat(8, minmax(0, 1fr)); --grid-9: repeat(9, minmax(0,

1fr)); --gr= id-10: repeat(10, minmax(0, 1fr)); --grid-11: repeat(11, minmax(0, 1fr)); -= -grid-12: repeat(12, minmax(0,

1fr)); --grid-page-width: var(--width-xl); -= -grid-page-gutter: 5vw; --grid-page-main: 2 / 3; --grid-page: minmax(var(--=

grid-page-gutter), 1fr) minmax(0, var(--grid-page-width)) minmax(var(--grid= -page-gutter), 1fr); } .gradio-container-5-

50-0-dev0 .prose { font-weight: var(--prose-text-weight= ); font-size: var(--text-md); } .gradio-container-5-50-0-

dev0 .prose \* { color: var(--body-text-color); } .gradio-container-5-50-0-dev0 .prose p { margin-bottom: var(--spacing-

sm); = line-height: var(--line-lg); } .gradio-container-5-50-0-dev0 .prose h1, .gradio-container-5-50-0-dev0 .pro= se

h2, .gradio-container-5-50-0-dev0 .prose h3, .gradio-container-5-50-0-de= v0 .prose h4, .gradio-container-5-50-0-

dev0 .prose h5 { margin: var(--spaci= ng-xxl) 0 var(--spacing-lg); font-weight: var(--prose-header-text-weight); = line-

height: 1.3; color: var(--body-text-color); } .gradio-container-5-50-0-dev0 .prose > :first-child { margin-top:

0px; } .gradio-container-5-50-0-dev0 .prose h1 { font-size: var(--text-xxl); } .gradio-container-5-50-0-dev0 .prose h2

{ font-size: var(--text-xl); } .gradio-container-5-50-0-dev0 .prose h3 { font-size: var(--text-lg); } .gradio-container-5-50-

0-dev0 .prose h4 { font-size: 1.1em; } .gradio-container-5-50-0-dev0 .prose h5 { font-size: 1.05em; } .gradio-container-

5-50-0-dev0 .prose ul { list-style: inside circle; } .gradio-container-5-50-0-dev0 .prose ol { list-style: inside

decimal; } .gradio-container-5-50-0-dev0 .prose ul > p, .gradio-container-5-50-0-dev0 = .prose li > p { display:

inline; } .gradio-container-5-50-0-dev0 .prose ol, .gradio-container-5-50-0-dev0 .pro= se ul { margin-top: 0px; padding-

left: 0px; } .gradio-container-5-50-0-dev0 .prose ul ul, .gradio-container-5-50-0-dev0 .= prose ul ol, .gradio-container-5-

50-0-dev0 .prose ol ol, .gradio-container-= 5-50-0-dev0 .prose ol ul { margin: 0.5em 0px; font-size: 90%; padding-inlin=

e-start: 2em; } .gradio-container-5-50-0-dev0 .prose li { margin-bottom: 0.5em; } .gradio-container-5-50-0-dev0 .prose

th, .gradio-container-5-50-0-dev0 .pro= se td { border-bottom: 1px solid rgb(225, 225, 225); padding: var(--text-xs= )

var(--text-md); text-align: left; } .gradio-container-5-50-0-dev0 .prose button, .gradio-container-5-50-0-dev0

= .prose .button, .gradio-container-5-50-0-dev0 .prose input, .gradio-contain= er-5-50-0-dev0 .prose textarea, .gradio-

container-5-50-0-dev0 .prose select= , .gradio-container-5-50-0-dev0 .prose fieldset { margin-bottom: var(--spac= ing-

sm); } .gradio-container-5-50-0-dev0 .prose pre, .gradio-container-5-50-0-dev0 .pr= ose blockquote, .gradio-container-5-

50-0-dev0 .prose dl, .gradio-container-= 5-50-0-dev0 .prose figure, .gradio-container-5-50-0-dev0 .prose table, .gra= dio-

container-5-50-0-dev0 .prose p, .gradio-container-5-50-0-dev0 .prose ul= , .gradio-container-5-50-0-dev0 .prose

ol, .gradio-container-5-50-0-dev0 .p= rose form { margin-bottom: var(--spacing-md); } .gradio-container-5-50-0-

dev0 .prose table, .gradio-container-5-50-0-dev0 .= prose tr, .gradio-container-5-50-0-dev0 .prose td, .gradio-container-5-

50-0= -dev0 .prose th { border: 1px solid var(--body-text-color); } .gradio-container-5-50-0-dev0 .prose table { border-

collapse: collapse; mar= gin-bottom: var(--spacing-xxl); } .gradio-container-5-50-0-dev0 .prose a { color: var(--link-text-

color); tex= t-decoration: underline; } .gradio-container-5-50-0-dev0 .prose a:visited { color: var(--link-text-col= or-

visited); } .gradio-container-5-50-0-dev0 .prose a:hover { color: var(--link-text-color= -hover); } .gradio-container-5-50-

0-dev0 .prose a:active { color: var(--link-text-colo= r-active); } .gradio-container-5-50-0-dev0 .prose hr { margin-top:

3em; margin-bottom: 3= .5em; border-width: 1px 0px 0px; border-top-style: solid; border-top-color:= rgb(225, 225,

225); } .gradio-container-5-50-0-dev0 .prose blockquote { border-left: 5px solid va= r(--border-color-primary); padding-

left: var(--size-2); margin: var(--size-= 6) 0 !important; } .gradio-container-5-50-0-dev0 .prose :last-child { margin-

bottom: 0px !impo= rtant; } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI----

Content-Type: text/css Content-Transfer-Encoding: quoted-printable Content-Location: cid:css-b643da76-c63b-4bf9-

acf3-ec971150c3d6@mhtml.blink @charset "utf-8"; @font-face { font-family: baidu-cos-icon; font-weight: 400; font-

style: nor= mal; src: url("chrome-extension://fhkhedgjlkcgdaldodbhmcldjglpabmf/pc-icon/= cos-icon/iconfont.woff")

format("woff"), url("chrome-extension://fhkhedgjlk= cgdaldodbhmcldjglpabmf/pc-icon/cos-icon/iconfont.woff2")

format("woff2"); } ------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-

Type: text/css Content-Transfer-Encoding: quoted-printable Content-Location: cid:css-afaa2172-4f03-47de-a895-

f73385f9ce49@mhtml.blink @charset "utf-8"; .monica-reading-highlight { animation: 1.5s ease-in-out 0s 1 normal none

ru= nning fadeInOut; } @keyframes fadeInOut {=20 0%, 100% { background-color: transparent; } 30%, 70%

{ background-color: rgba(2, 118, 255, 0.2); } } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/css Content-Transfer-Encoding:

quoted-printable Content-Location: cid:css-ef86ea64-e17f-4267-8589-f5672a634ce6@mhtml.blink @charset "utf-

8"; :root { --bg: white; --col: #27272a; --bg-dark: #0f0f11; --col-dark: #f4f4f= 5; } body { background: var(--bg); color:

var(--col); font-family: Arial, Helvet= ica, sans-serif; } @media (prefers-color-scheme: dark) { body { background:

var(--bg-dark); color: var(--col-dark); } } ------MultipartBoundary--

a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI---- Content-Type: text/html Content-ID:

Content-Transfer-

Encoding: quoted-printable

------MultipartBoundary--a3YPeNu6ajym40DyOnBO6VjTBCUxhAS64NLrFe3NvI------