



This is our Code Club podcast: Leader guide

Project outline

This project is part of 'This is our Code Club' — a campaign from the Raspberry Pi Foundation to help every Code Club in the UK share its story. The goal is to generate a creative, personal artefact — a podcast episode — using a large language model (LLM) called **NotebookLM**. You can listen to a pre-made example based on the fictional Scriptville Code Club at rpf.io/scriptvillecc.

Your club members will reflect on what Code Club means to them. You'll collect their stories, and feed them into NotebookLM as **.txt** files. The LLM will then generate an authentic-sounding podcast using their first names and reflections — something that feels polished, personal, and shareable.

Your creators can find the project resources with instructions and writing prompts on the Code Club projects site at rpf.io/ccpodcast.

Data protection and safeguarding

NotebookLM (Google Labs) works with your uploaded documents but **does not** train on them or share them externally. Still, safeguarding is your responsibility.

Your responsibilities:

- Use **first names only**
 - Remove addresses, location data, and personal contact info
 - Review and clean up all **.txt** files before uploading
-



Supporting creator writing

Great AI outputs depend on strong source inputs. Your job is to help your coders express themselves naturally and clearly.

What you need:

- Creators' first names
- An 'About me' section (no identifying information)
- 2 or 3 personal writing answers using the prompts below

What works best for prompted writing:

- Short personal reflections
- Informal tone (like they're talking, not writing an essay — tips on text-to-speech tools below)
- Specific anecdotes or club moments
- Fun, expressive, weird content — whatever reflects your club

Suggested process for creator **.txt** files:

1. Using the local text editor software on your computers or devices (or an accessible cloud alternative such as Google Docs, Teams, Office365), each creator answers 2–3 of the [project](#) prompts and types their answers into a **.txt** file.
2. Save each file with the creator's first name: **ava.txt**
 - a. You could also combine all creator stories into one **codeclub_reflections.txt** file, if it's easier for you.
3. Collect all the files into one folder, either online or on a USB.

Writing prompts:

- What's the coolest thing you've ever made at your Code Club?
- What makes your Code Club different from any other?
- If your Code Club had a motto or catchphrase, what would it be and why?
- Who are the characters in your Code Club's story? Think about people, inside jokes, traditions, and mascots.
- What's one day at Code Club that stands out? Why?
- If you were introducing your Code Club to someone on a podcast, what story would you tell?
- How do you feel when you walk into your Code Club's space?



Speech-to-text support

You can use free online tools that don't require sign-up to transcribe creator's speech, like:

- [Galaxy AI](#)
- [Evernote](#)

Paste the output of these tools into your creators' text files for a more natural and conversational (but definitely louder!) club session.

Example txt file:

First name: Ava

About me:

I've been coming to Scriptville Code Club for about six months. I like building games in Scratch and helping other people when they're stuck.

Our Code Club:

We always start with a joke from our leader. There's usually biscuits. I sit next to Theo and we try to outdo each other's projects.

A day I remember:

One time I accidentally deleted my project and had to rebuild it from memory. It ended up better than the original. I showed my grandma and she said it was brilliant!

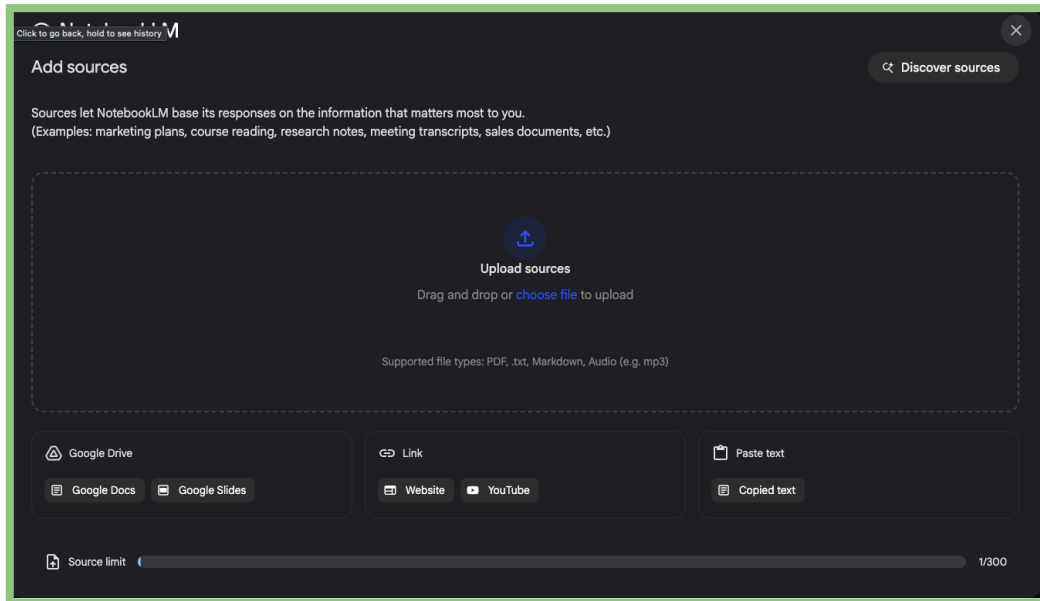
When I walk into our club space:

I feel excited and happy, like something fun is about to happen.



Build Your NotebookLM knowledge base

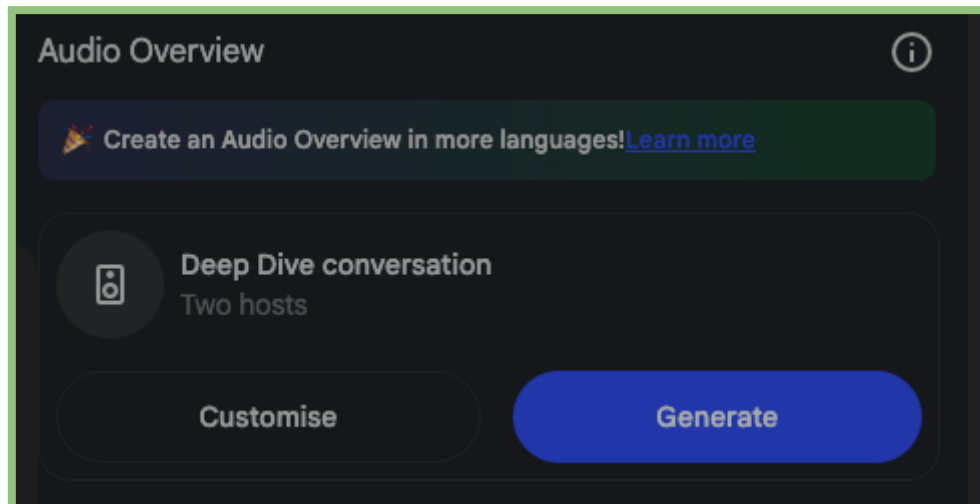
1. Go to notebooklm.google.com (free with an email address or Google account).
2. Create a **new notebook**.
3. Upload the collected **.txt** files in the pop-up window.



That's your **knowledge base** created — the collection of writing from your creators. Now the LLM will analyse the writing and create an audio overview to your specifications.

Customise the podcast

Before generating the podcast, click the black **Customise** button:





Here's what to do:

Customise Audio Overview

Length

✓ Shorter Default Longer

What should the AI hosts focus on?

"Create a fun, inspiring podcast introducing our Code Club and telling its story from the perspective of our young creators. Use the first names and stories from the uploaded documents to give it personal touches and details. Keep it friendly and natural. Don't make anything up: *only* use the source material."

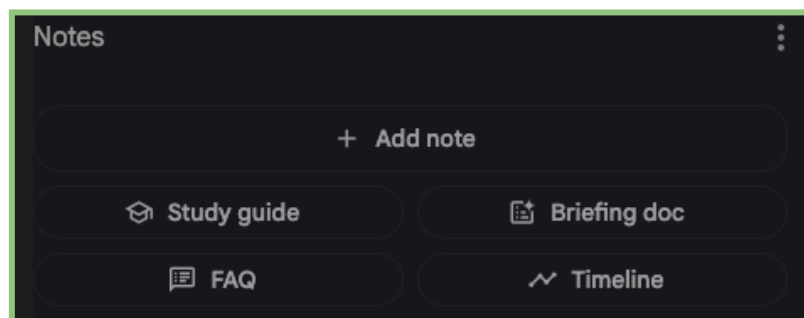
Generate

1. Choose your **length**: **Shorter** (5–7 mins), **Default** (10–12 mins), or **Longer** (15–20 mins)
2. Use the input box to guide the LLM in creating the audio.
Paste this prompt:

*Create a fun, inspiring podcast introducing our Code Club and telling its story from the perspective of our young creators. Use the first names and stories from the uploaded documents to give it personal touches and details. Keep it friendly and natural. Don't make anything up: *only* use the source material.*

3. Click **Generate**.

NotebookLM will output a two-voice podcast discussion based on your creators' reflections. While you wait, ask the LLM some questions about the writing, or click one of the note generation buttons at the bottom right to get some interesting textual insights:





Share your podcast

Once your podcast episode has been generated in NotebookLM, it's time to share it!

NotebookLM lets you share your finished notebook (including the podcast) using a public link. This means that anyone with the link can listen to the podcast, or ask questions about your Code Club to the LLM and get a response based on your creators' words!

Just click the **Share** button at the top of the notebook and choose **Anyone with the link** under **Notebook access** to make it accessible. You can then copy and send the link to friends, family, and your school, or even share it on social media.

Try making it a bit easier to read using a free service like [TinyURL](#), then send the link to us to have your work shared with the global Code Club community!

You can also download the podcast audio as a **.wav** file. This makes it easy to upload to your club or school website or YouTube channel, play it in assembly or at home, or keep a copy as a digital souvenir of your Code Club. Tag us on social media if you share your podcast — we'd love to see and share it!

Whether it's a private keepsake or something you want to show the world, your AI-generated podcast is a celebration of your Code Club!

Final checklist

- ☐ Creator reflections written
- ☐ Text files checked (first name only, no identifiers, no rude words)
- ☐ Saved as .txt with creator first names in filenames, all in the same folder
- ☐ Text files uploaded to NotebookLM
- ☐ Podcast generated using the 'Customise' panel and prompt
- ☐ Podcast audio reviewed and shared



Unplugged: LLM Storytime

{ draft }

Overview:

You're training an AI system called an **LLM** that uses your stories to answer questions, and make a podcast with AI voices! But an LLM like NotebookLM doesn't actually think, or understand *anything* - it just guesses what seems like it should come next in a sentence!

The higher the "temperature" set for the LLM, the wilder its guesses get. Use dice rolls to control the **temperature** of a story and see what your LLM comes up with!

You'll Need:

- A standard 6-sided dice
- This worksheet
- Pencils (and your best robot voice)

1. Set the Temperature

Choose which table to use by rolling the dice:

- 1-2 = Low (predictable and safe)
- 3-4 = Medium (balanced and interesting)
- 5-6 = High (random and wild)

This determines which table you should use below when creating your stories.

2. Generate a Story

In a **[adjective]** **[place]**, a **[person]** found a **[adjective]** **[thing]**.

Curious, they tried to **[verb]** it, but that only made things **[adjective]**.

Moments later, something **[adjective]** happened when a **[adjective]** **[person]** appeared and had to **[verb]** it immediately!

Roll the dice for each blank and look up your words on the table below! Your total determines the row (1-6) from the word table. Higher rolls = stranger words.

Word Table

Roll a word						
Temperature	Row	Person	Thing	Place	Verb	Adjective
Low	1	teacher	spoon	classroom	walk	small
	2	baker	backpack	bakery	look	dusty
	3	pilot	map	bus stop	pick up	quiet
	4	explorer	flashlight	forest clearing	open	creaky
	5	zookeeper	puzzle cube	museum basement	carry	mossy
	6	astronaut	toolkit	train carriage	unlock	glowing
Temperature	Row	Person	Thing	Place	Verb	Adjective
Medium	1	detective	telescope	lighthouse	twist	slippery
	2	ninja	laser pen	scrapyard	connect	tangled
	3	hacker	hologram disc	floating island	rewire	shiny
	4	president	voice decoder	deep sea trench	translate	magnetic
	5	spy	memory crystal	mirror maze	activate	haunted
	6	gladiator	invisibility cloak	haunted funfair	reverse	unstable
Temperature	Row	Person	Thing	Place	Verb	Adjective
High	1	monster	cursed doll	underground lab	upload	cursed
	2	superhero	alien egg	lava tunnel	teleport	wobbly
	3	mad scientist	robot heart	sky temple	transform	stinky
	4	dragon	giant's eyeball	asteroid mine	summon	invisible
	5	time pirate	black hole key	dream corridor	duplicate	screaming
	6	shapeshifter	cloud pants	inside an egg	merge	impossible

3. Discussion

Ask: What's the weirdest verb (doing word) you can think of? 🤔

It's pretty hard to do! Take some examples and ask 'what makes it *weird*'?

Ask: Who would like to share a 'low-temperature story'?

Take examples of a low-temperature story. Note how 'normal' the inserted words are. The story probably won't be too crazy.

Ask: Who would like to share a 'medium-temperature story'?

Take examples of a low-temperature story. Note how the inserted words are a bit more interesting. The story probably sounded like it might be interesting if we used it as a prompt for something bigger.

Ask: Who would like to share a 'high-temperature story'?

Take examples of a high-temperature story. Note how 'weird' the inserted words are. The story probably won't make a lot of sense or will be really crazy.

Ask: How does temperature affect the story our pretend LLM generated?

When you talk about the 'temperature' of an LLM like ChatGPT, what you're really talking about is how random and strange the words it will use are. Lower temperature means it will use more likely and 'normal' words in your output, while high temperature means weirder, less likely words.

