

Hands-on Workshop (Instructions)

QPulse — Learner Setup & Runbook (Step-by-Step)

Goal: set up QPulse end-to-end and run a live pulse in a Telegram group.

What you will build

- A form that creates an active question
- A Telegram bot that accepts responses from participants
- A results workflow that aggregates answers and produces a summary report

Prerequisites

- n8n (Cloud)
- Telegram account
- OpenAI API key
- Workflow JSON files: AddQuestion.json, UserResponse.json, GetResults.json

Step 1 — Create a Telegram bot

- Open Telegram → @BotFather → /newbot → follow prompts to create a bot.
- Copy the bot token (you will paste this into n8n credentials).
- Add the bot to the Telegram group where participants will respond.

Step 2 — Start n8n

- Create a n8n trial account
- Open n8n Cloud (if you already use it)

Step 3 — Create credentials in n8n

- Telegram credential: paste bot token.
- OpenAI credential: paste API key.

Step 4 — Create Data Tables

Create two data tables in n8n (names and columns must match):

- QPulse_Questions: question (string), summaryDone (boolean)
- QPulse_Answers: userId (string), question (string), answer (string)

Tip: keep column IDs exactly the same (case-sensitive).

Step 5 — Import the workflows

- Workflows → Import from File → import AddQuestion.json
- Import UserResponse.json
- Import GetResults.json

Or Build it live in class.

Step 6 — Configure each workflow

A) AddQuestion

- Confirm Form Trigger has one field for the question.
- Confirm Insert row writes question + summaryDone=false into QPulse_Questions.

B) UserResponse (Telegram)

- Set Telegram Trigger to your Telegram credential.
- Set OpenAI Chat Model to your OpenAI credential.
- Verify tools: Get_QPulse_Question reads QPulse_Questions where summaryDone=false.
- Verify tools: Insert to QPulse_Answers writes userId + question + answer into QPulse_Answers.
- Verify Send a text message replies in Telegram.

C) GetResults (Summarize)

- GateKeeper must have GetQuestion tool connected and SOP_Question output parser enabled.
- Retriever must have get_QPulse_Answers tool connected and Structured Output Parser enabled.
- Summarizer returns final output and Respond to Chat delivers it.

Step 7 — Activate workflows

- Activate UserResponse (Telegram Trigger). This is required.
- Activate AddQuestion if learners will use the form live.
- Activate GetResults if you will trigger summarization from the chat trigger UI.

Step 8 — Run the project (live demo script)

- Open AddQuestion form URL → submit a question (e.g., “Where should we go for lunch?”).
- In Telegram group: participants greet the bot and respond to the question.

- Run GetResults: in n8n's chat trigger UI send “get results” or “summarize”.
- Review the output: winner + counts table + summary.

Troubleshooting

- Bot doesn't respond: ensure the workflow is active and the bot is in the group.
- No open question: confirm QPulse_Questions has a row with summaryDone=false.
- Answers not stored: verify Insert to QPulse_Answers mappings for userId/question/answer.
- Results empty: ensure answers' question text matches the question stored in QPulse_Questions.
- Parser errors: ensure the LLM outputs match the schema (question + answers[]).

Extensions (optional)

- Set summaryDone=true after a successful summarization to prevent repeats.
- Add time-limit closing and ignore late responses.
- Add word cloud + sentiment distribution.
- Add anonymity: hash userId or omit it entirely.