Activity #2:
Highlight what each evaluator is evaluating.

Response
All three of these are LangSmith "LLM as judge" evaluators (based on this parameter: config={"llm" : eval_llm}

In our case, using gpt-4.1-mini based on this code:

```
from langchain_openai import ChatOpenAI
llm = ChatOpenAI(model="gpt-4.1-mini")
```

qa_evaluator:
This is a LangSmith built-in evaluator. It is intended to evaluate the accuracy of the model, based on the quality of the output compared to a known good answer (aka reference or ground truth).
This evaluator requires that there be a reference answer provided. It uses an LLM to compare the system's result (aka prediction) to the reference answer.

labeled_helpfulness_evaluator:
This is a semi-custom evaluator, calling LangChainStringEvaluator with the "labeled_criteria" identifier.
It is customized based on the criteria defined in the config dictionary.

In our code the "labeled criteria" evaluator is configured to measure "helpfulness", based on the criteria prompt provided:
"Is this submission helpful to the user, taking into account the correct reference answer?"

Similar to "qa", above, this evaluator also requires a reference answer, to compare with the system output.

Alternatively, if reference answers were not available, we could configure a more open ended "helpfulness" evaluator by omitting the last sentence of the prompt.

Since we do have reference answers from our SDG (using GPT-4.1), we can use this somewhat more grounded way to evaluate helpfulness.

empathy_evaluator:
This is a fully customized evaluator, calling LangChainStringEvaluator with the "criteria" identifier.

The criteria here is defined as "empathy", based on the criteria prompt:
"Is this response empathetic? Does it make the user feel like they are being

heard?"

Notably, this evaluator does not require a reference answer or ground truth, since the evaluation is not comparing the output to any reference.