To format an answer to the original question based on the given prompt, we should look at the output of the tool calls that were made.

The final ToolMessage has content 'PDF written successfully to user_question_response.pdf!'. This indicates that a PDF was created with the output of a tool call for this message. However, it does not give us any information about what function or arguments were used in the tool call that generated this PDF.

Looking at the prompt_eval_metadata, we see there is a 'tool_calls' list where each item has a 'function' dictionary which contains a name and an 'arguments' object. There are two items in the 'tool_calls' list.

The first one has:
```
{
   "name": "Create_PDF_tool",
   "arguments": {
      "file_name": "user_question_response.pdf",
      "scratchpad": "The user asked: what if i multiply 3 with 12.\n\nIntermediate reasoning: We can use the 'multiply' tool to solve this question.\n\nFinal answer: The final answer is $36.0.",
      "text": "The final answer is $\boxed{36.0}$"
   },
   ...
}
```

This indicates that a Create_PDF_tool was called with arguments for file_name, scratchpad and text.

However, looking at the original prompt, it only mentions one tool call which is not Create_PDF_tool but rather multiply.

Hence, we should ignore this item as it seems unrelated to the original question.

The second item in 'tool_calls' has:
```
{
    "name": "Create_PDF_tool",
    "arguments": {
        "file_name": "user_question_response.pdf",
        "scratchpad": "The user asked: what if i multiply 3 with 12.\n\nIntermediate reasoning: We can use the 'multiply' tool to solve this question.\n\nFinal answer: The final answer is $36.0.",
        "text": "The final answer is $\boxed{36.0}$"
    },
    ...
}
```
This item also seems unrelated as it again has the Create_PDF_tool which we previously found to be unrelated.

However, looking at the prompt_eval_metadata again, there's a 'tool_call_chunks' list with two items.

One of them has:

```
{
    "name": "multiply",

    ...

}
```

So it looks like what was called initially in this message was indeed the multiply function.

Agent Scratchpad: