

## My computer:

Operating system - macOS Sequoia

Chip - Apple M1

Memory - 8 GB

## Environment:

```
azamat@MacBook-Pro ~ % git --version
git version 2.39.5 (Apple Git-154)
azamat@MacBook-Pro ~ % javac --version
javac 21.0.4
azamat@MacBook-Pro ~ % mvn --version
Apache Maven 3.9.9 (8e8579a9e76f7d015ee5ec7bfcdc97d260186937)
Maven home: /Users/azamat/.sdkman/candidates/maven/current
Java version: 21.0.4, vendor: Oracle Corporation, runtime: /Users/azamat/.sdkman/candidates/java/21.0.4-oracle
Default locale: en_US, platform encoding: UTF-8
OS name: "mac os x", version: "15.0.1", arch: "aarch64", family: "mac"
azamat@MacBook-Pro ~ % make --version
GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for i386-apple-darwin11.3.0
azamat@MacBook-Pro ~ % python --version
Python 3.12.7
azamat@MacBook-Pro ~ % docker --version

Docker version 27.2.0, build 3ab4256
azamat@MacBook-Pro ~ % █
```

## Ollama

```
azamat@MacBook-Pro ~ % ollama -v
ollama version is 0.3.13
azamat@MacBook-Pro ~ % ollama list
NAME          ID          SIZE    MODIFIED
gemma2:2b     8ccf136fdd52 1.6 GB  4 minutes ago
azamat@MacBook-Pro ~ % █
```

## Ollama API:

```
azamat — zsh — 175x21
~ — ollama run gemma2:2b
~ — -zsh
azamat@MacBook-Pro ~ % curl http://localhost:11434/api/generate -d '{ "model": "gemma2:2b", "prompt": "What are you?", "stream":false }'
{"model":"gemma2:2b","created_at":"2024-10-14T16:33:35.255475Z","response":"I am Gemma, an open-weight AI assistant. Think of me as a computer program that can understand and generate human language. I'm still under development, but I have learned to perform many kinds of tasks, like:\n\n**Answering questions:** I can provide summaries of factua\nl topics or create stories.\n**Generating text:** I can write different kinds of creative content, from poems to scripts. \n\n**Translating languages:** I can help you und\nerstand text in another language. \n\nIs there anything specific you'd like me to try? 😊 \n","done":true,"done_reason":"stop","context":[106,1645,108,1841,708,692,235336,107,\n108,106,2516,108,235285,1144,137061,235269,671,2174,235290,7553,16481,20409,235265,139,31008,576,682,685,476,6875,2733,674,798,3508,578,11941,3515,5255,235265,590,235303,23526\n2,2076,1362,3505,235269,981,590,791,9710,577,3114,1767,13049,576,13333,235269,1154,235292,109,235287,5231,1261,574,3920,66058,590,798,3658,97361,576,63914,16254,689,3104,8965,\n235265,108,235287,5231,136618,2793,66058,139,235285,798,5598,2167,13049,576,12354,3381,235269,774,29632,577,37089,235265,235248,108,235287,5231,3593,80937,17044,66058,139,2352\n85,798,1707,692,3508,2793,575,2550,5255,235265,235248,109,2437,1104,4341,3724,692,235303,235258,1154,682,577,3418,235336,44416,235248,1081,"total_duration":5126501459,"load_du\nration":59720625,"prompt_eval_count":13,"prompt_eval_duration":87807000,"eval_count":123,"eval_duration":4977769000}"}
azamat@MacBook-Pro ~ %
```

## Ollama API And jq:

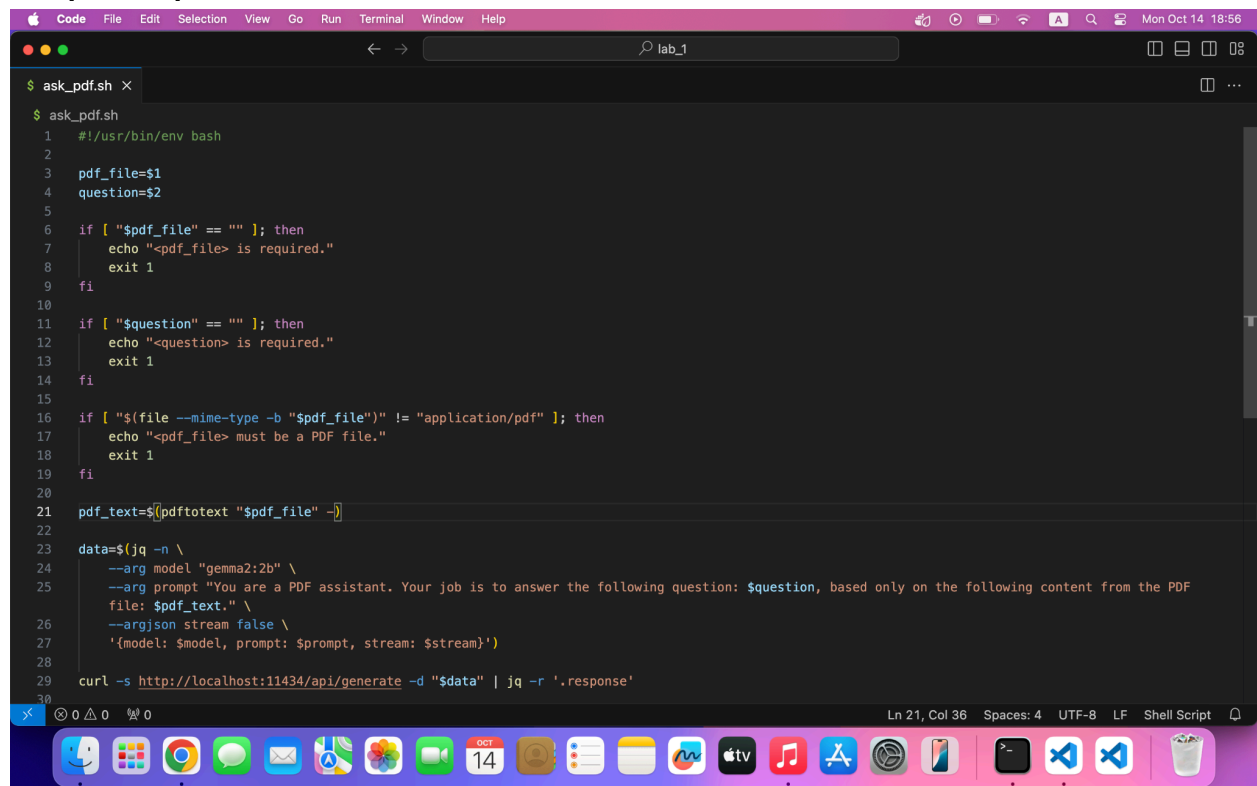
```
azamat — zsh — 175x21
~ — ollama run gemma2:2b
~ — -zsh
azamat@MacBook-Pro ~ % curl http://localhost:11434/api/generate -d '{ "model": "gemma2:2b", "prompt": "What are you?", "stream":false }' | jq '.response'
{
  "model": "gemma2:2b",
  "created_at": "2024-10-14T16:33:35.255475Z",
  "response": "I am Gemma, an open-weight AI assistant. Think of me as a computer program that can understand and generate human language. I'm still under development, but I have learned to perform many kinds of tasks, like:\n\n**Answering questions:** I can provide summaries of factua\nl topics or create stories.\n**Generating text:** I can write different kinds of creative content, from poems to scripts. \n\n**Translating languages:** I can help you und\nerstand text in another language. \n\nIs there anything specific you'd like me to try? 😊 \n",
  "done": true,
  "done_reason": "stop",
  "context": [106,1645,108,1841,708,692,235336,107,108,106,2516,108,235285,1144,137061,235269,671,2174,235290,7553,16481,20409,235265,139,31008,576,682,685,476,6875,2733,674,798,3508,578,11941,3515,5255,235265,590,235303,235262,2076,1362,3505,235269,981,590,791,9710,577,3114,1767,13049,576,13333,235269,1154,235292,109,235287,5231,1261,574,3920,66058,590,798,3658,97361,576,63914,16254,689,3104,8965,235265,108,235287,5231,136618,2793,66058,139,235285,798,5598,2167,13049,576,12354,3381,235269,774,29632,577,37089,235265,235248,108,235287,5231,3593,80937,17044,66058,139,235285,798,1707,692,3508,2793,575,2550,5255,235265,235248,109,2437,1104,4341,3724,692,235303,235258,1154,682,577,3418,235336,44416,235248,1081],
  "total_duration": 5126501459,
  "load_duration": 59720625,
  "prompt_eval_count": 13,
  "prompt_eval_duration": 87807000,
  "eval_count": 123,
  "eval_duration": 4977769000
}
azamat@MacBook-Pro ~ %
```

## XPDF toolkit for pdf text extraction:

```
azamat — zsh — 175x38
~ — ollama run gemma2:2b
~ — -zsh
azamat@MacBook-Pro ~ % pdftotext -h
pdftotext version 4.05 [www.xpdfreader.com]
Copyright 1996-2024 Glyph & Cog, LLC
Usage: pdftotext [options] <PDF-file> [<text-file>]
  -f <int>           : first page to convert
  -l <int>           : last page to convert
  -layout            : maintain original physical layout
  -simple            : simple one-column page layout
  -simple2           : simple one-column page layout, version 2
  -table            : similar to -layout, but optimized for tables
  -lineprinter      : use strict fixed-pitch/height layout
  -raw             : keep strings in content stream order
  -fixed <number>   : assume fixed-pitch (or tabular) text
  -linespacing <number> : fixed line spacing for LinePrinter mode
  -clip            : separate clipped text
  -nodiag          : discard diagonal text
  -enc <string>     : output text encoding name
  -oel <string>     : output end-of-line convention (unix, dos, or mac)
  -noppbrk         : don't insert a page break at the end of each page
  -bom             : insert a Unicode BOM at the start of the text file
  -marginl <number> : left page margin
  -marginr <number> : right page margin
  -marginl <number> : top page margin
  -marginb <number> : bottom page margin
  -opw <string>     : owner password (for encrypted files)
  -upw <string>     : user password (for encrypted files)
  -verbose         : print per-page status information
  -q              : don't print any messages or errors
  -cfg <string>     : configuration file to use in place of .xpdfrc
  -listencodings   : list all available output text encodings
  -v              : print copyright and version info
  -h              : print usage information
  -help           : print usage information
  -?             : print usage information
azamat@MacBook-Pro ~ %
```

the content of the provided PDF document.

## Ask pdf script:

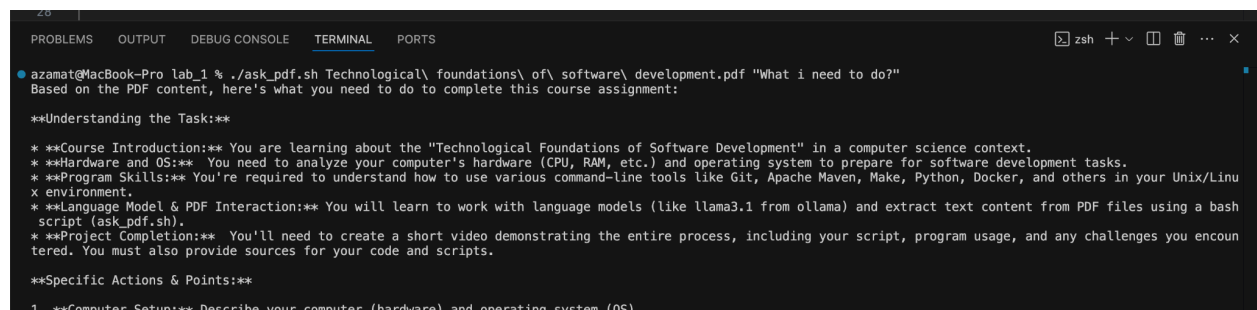


The screenshot shows a macOS Code editor window with a dark theme. The menu bar at the top includes 'Code', 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', 'Window', and 'Help'. The title bar shows 'lab\_1'. The editor displays a shell script named 'ask\_pdf.sh' with the following content:

```
$ ask_pdf.sh x

$ ask_pdf.sh
1  #!/usr/bin/env bash
2
3  pdf_file=$1
4  question=$2
5
6  if [ "$pdf_file" == "" ]; then
7      echo "<pdf_file> is required."
8      exit 1
9  fi
10
11 if [ "$question" == "" ]; then
12     echo "<question> is required."
13     exit 1
14 fi
15
16 if [ "$(file --mime-type -b "$pdf_file")" != "application/pdf" ]; then
17     echo "<pdf_file> must be a PDF file."
18     exit 1
19 fi
20
21 pdf_text=$(pdftotext "$pdf_file" -)
22
23 data=$(jq -n \
24     --arg model "gemma2:2b" \
25     --arg prompt "You are a PDF assistant. Your job is to answer the following question: $question, based only on the following content from the PDF
26     file: $pdf_text." \
27     --argjson stream false \
28     '{model: $model, prompt: $prompt, stream: $stream}')
29
30 curl -s http://localhost:11434/api/generate -d "$data" | jq -r '.response'
```

The status bar at the bottom indicates 'Ln 21, Col 36', 'Spaces: 4', 'UTF-8', 'LF', and 'Shell Script'.



The screenshot shows a VS Code terminal window with a dark theme. The terminal displays the output of the 'ask\_pdf.sh' script, which is a response from a language model. The output is as follows:

```
azamat@MacBook-Pro lab_1 % ./ask_pdf.sh Technological\ foundations\ of\ software\ development.pdf "What i need to do?"
Based on the PDF content, here's what you need to do to complete this course assignment:

**Understanding the Task:**

* **Course Introduction:** You are learning about the "Technological Foundations of Software Development" in a computer science context.
* **Hardware and OS:** You need to analyze your computer's hardware (CPU, RAM, etc.) and operating system to prepare for software development tasks.
* **Program Skills:** You're required to understand how to use various command-line tools like Git, Apache Maven, Make, Python, Docker, and others in your Unix/Linux environment.
* **Language Model & PDF Interaction:** You will learn to work with language models (like llama3.1 from ollama) and extract text content from PDF files using a bash script (ask_pdf.sh).
* **Project Completion:** You'll need to create a short video demonstrating the entire process, including your script, program usage, and any challenges you encountered. You must also provide sources for your code and scripts.

**Specific Actions & Points:**

1. **Computer Setup:** Describe your computer (hardware) and operating system (OS).
```