edunet foundation

LAB MANUAL

Deploying Web Applications with Streamlit Community Cloud and GitHub

Application 1: Student Marks and Grades Calculator

1. Objective

To build a web application that allows users (faculty or students) to input marks for multiple subjects and automatically calculates total marks, percentage, and assigns a grade. This helps demonstrate how Streamlit can be used for simple, interactive academic tools.

2. Equipment Required

- A computer or laptop with internet connectivity
- Python (3.9 or above) installed
- VS Code or any code editor
- Web browser (Google Chrome/Edge)
- GitHub account
- Streamlit Community Cloud account

3. Prerequisites

- Basic knowledge of Python programming (variables, loops, conditionals)
- Familiarity with using pip to install Python libraries
- Basic understanding of GitHub usage (optional for deployment stage)

4. Problem Statement

Manually calculating percentages and grades for multiple students or subjects can be time-consuming and error-prone. The objective is to create a Streamlit-based interactive web application where the user can input marks for subjects and instantly get the total marks, percentage, and grade without manual calculations.

5. Procedure

a. Create a new folder for the project.

b. Create a Python file named streamlit_app.py.

c. Write the Streamlit code to:

- Accept number of subjects and marks as input.
- Calculate total, percentage, and assign a grade.
- Display results interactively.

d. Create a requirements.txt file containing: streamlit

e. Push the code to a GitHub repository.

f. Deploy the app using Streamlit Community Cloud.

6. **Setting up the Environment**
    1. Install Python (if not already installed).
    2. Install Streamlit ***(in cmd)***: pip install streamlit
    3. Verify installation: streamlit hello
    4. Open a code editor (VS Code or similar) and open your project folder.

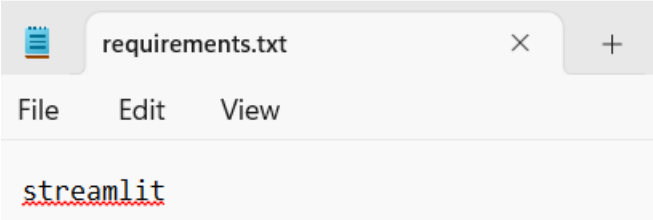
Steps:

Perform the following steps on the local machine

1. Create a new folder. We will add the python code file and the requirements file in this folder.
2. Create a new python file. Here we are calling it streamlit_app.py

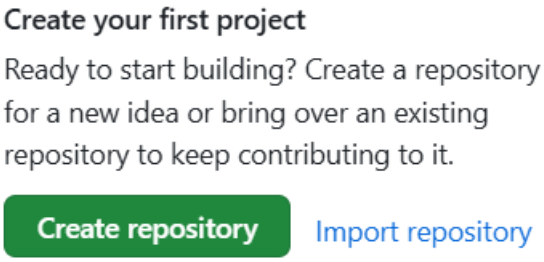
```
import streamlit as st

st.title("Student Marks & Grade Calculator")

subjects = st.number_input("Enter number of subjects", min_value=1, max_value=10,
value=5)
marks = []

for i in range(int(subjects)):
    marks.append(st.number_input(f"Enter marks for subject {i+1} (out of 100)",
min_value=0, max_value=100))

if st.button("Calculate Result"):
    total = sum(marks)
    percentage = total / (subjects * 100) * 100
    st.write(f"Total Marks: {total}")
    st.write(f"Percentage: {percentage:.2f}%")

    if percentage >= 90:
        st.success("Grade: A+")
    elif percentage >= 75:
        st.info("Grade: A")
    elif percentage >= 60:
        st.info("Grade: B")
    elif percentage >= 50:
        st.warning("Grade: C")
    else:
        st.error("Grade: Fail")
```

3. Create a new requirements.txt file.

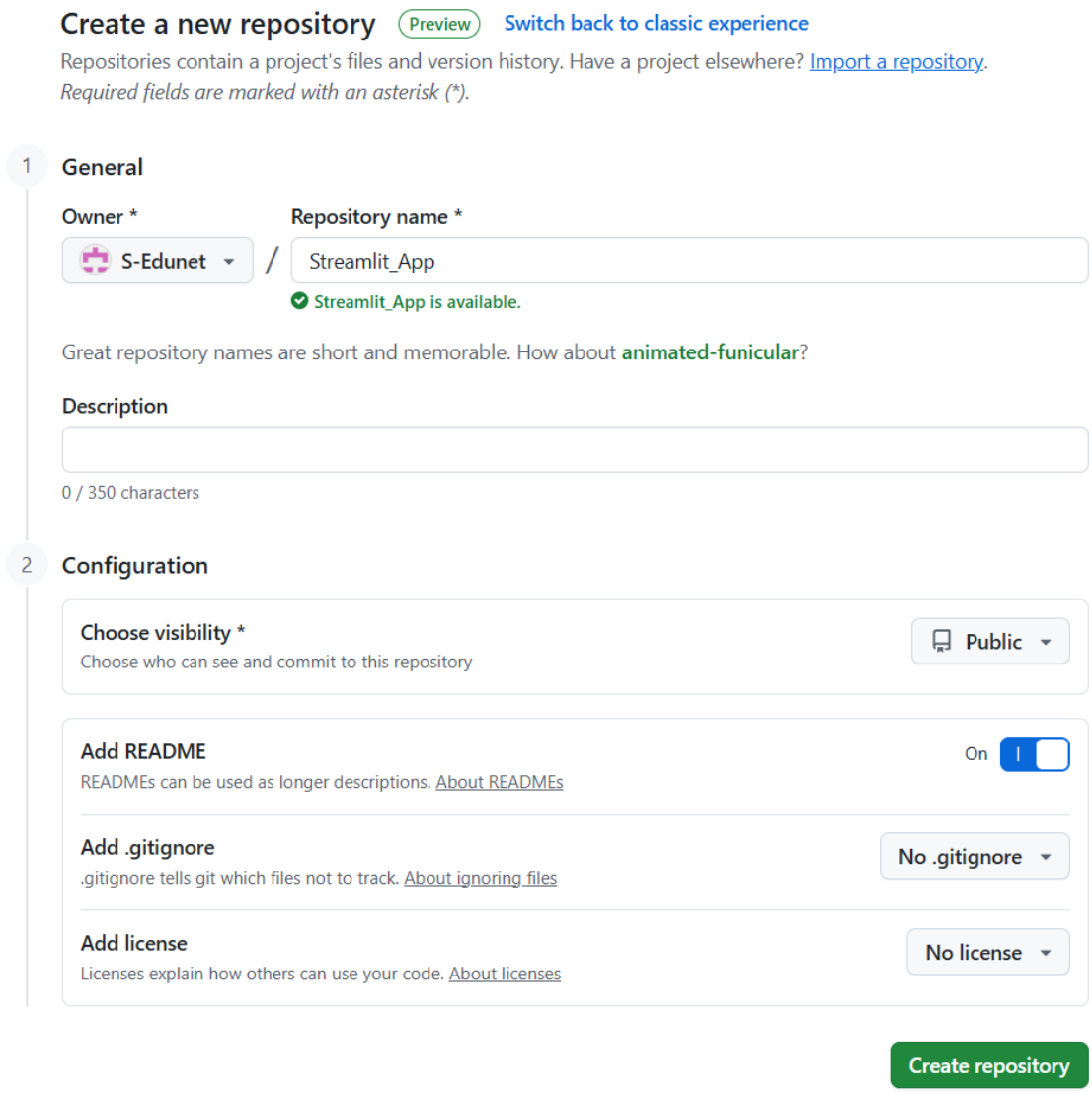


Perform the following steps on GitHub

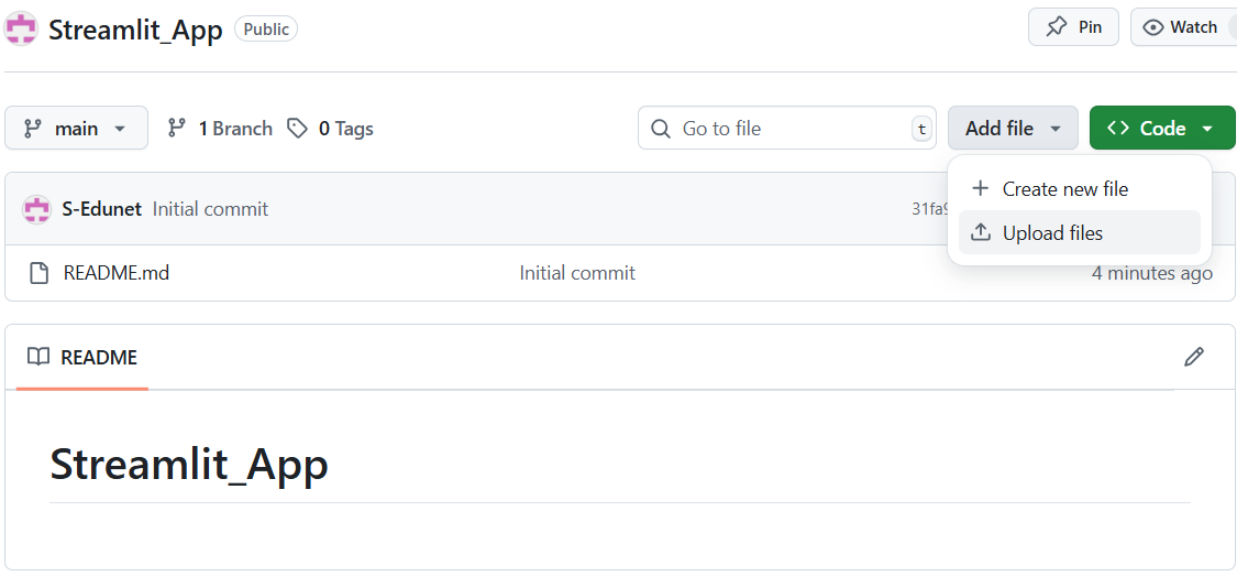
1. **Navigate to GitHub:** https://github.com/ and sign in using the previouly created account.
2. **Create a new repository**



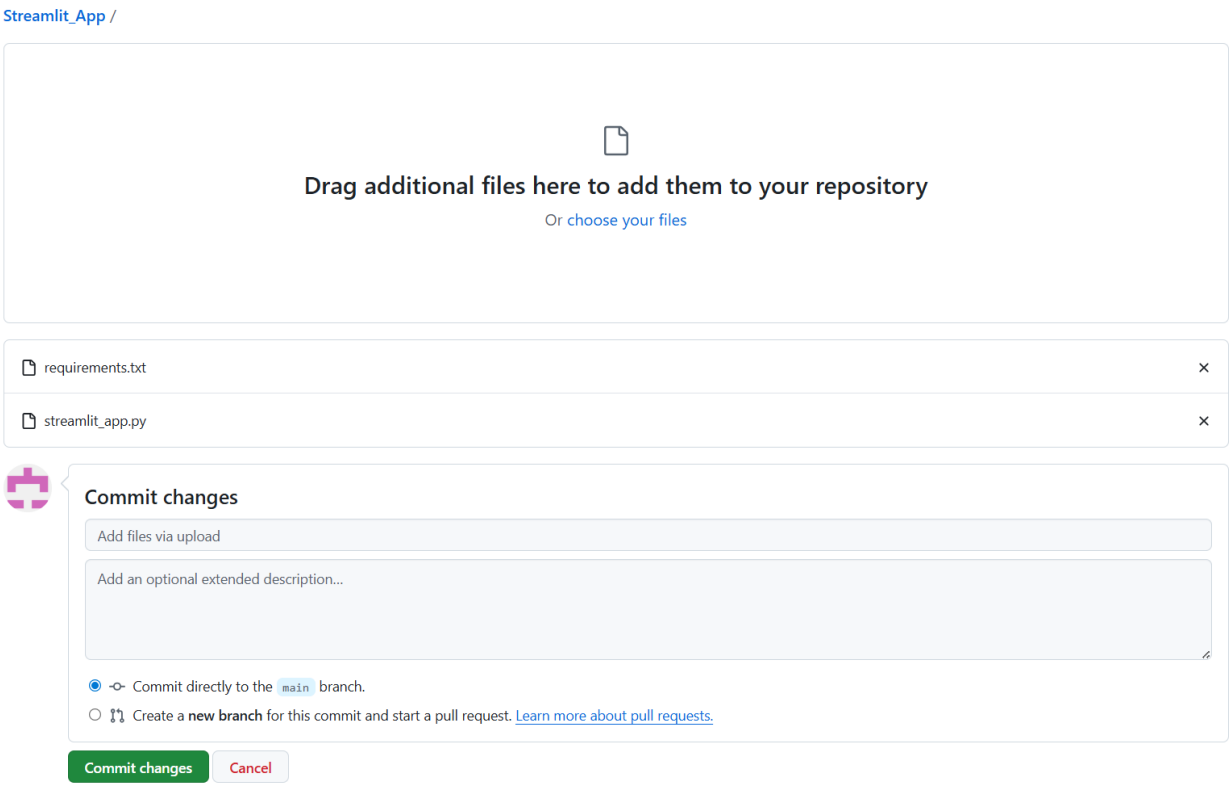
3. **Assign a name to the repository, click on the toggle button for README file, and click on Create Repository**



4. **Click on Add File and select Upload Files.**
   a. Navigate to the folder where the python file and the requirements file are stored and upload them.

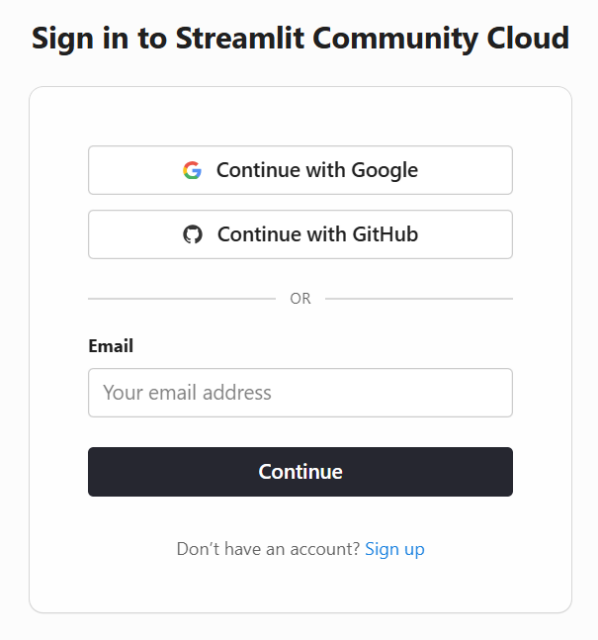


5. **After the files are uploaded, click on Commit changes**



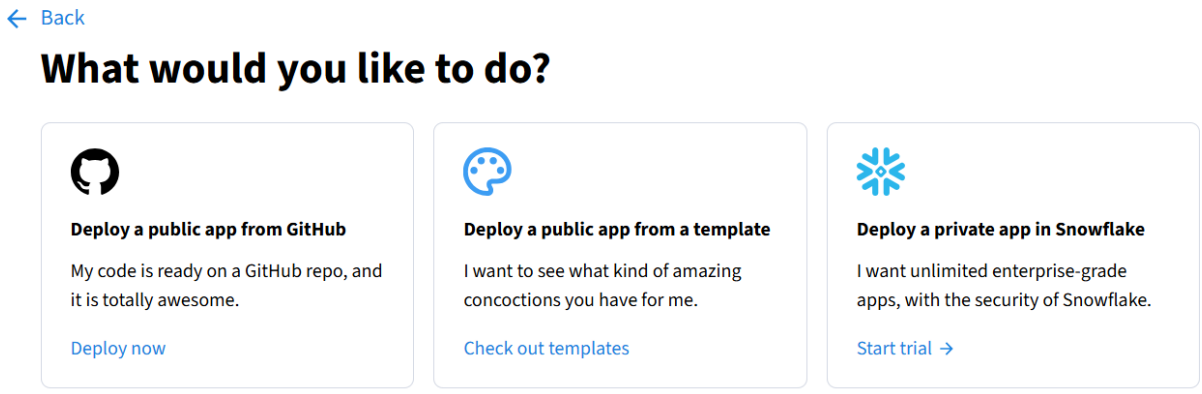
Perform the following steps on Strealmit

1. **Navigate to https://streamlit.io/cloud and login your account OR create a new account. However it is recommended that you sign in usin th GitHub account.**

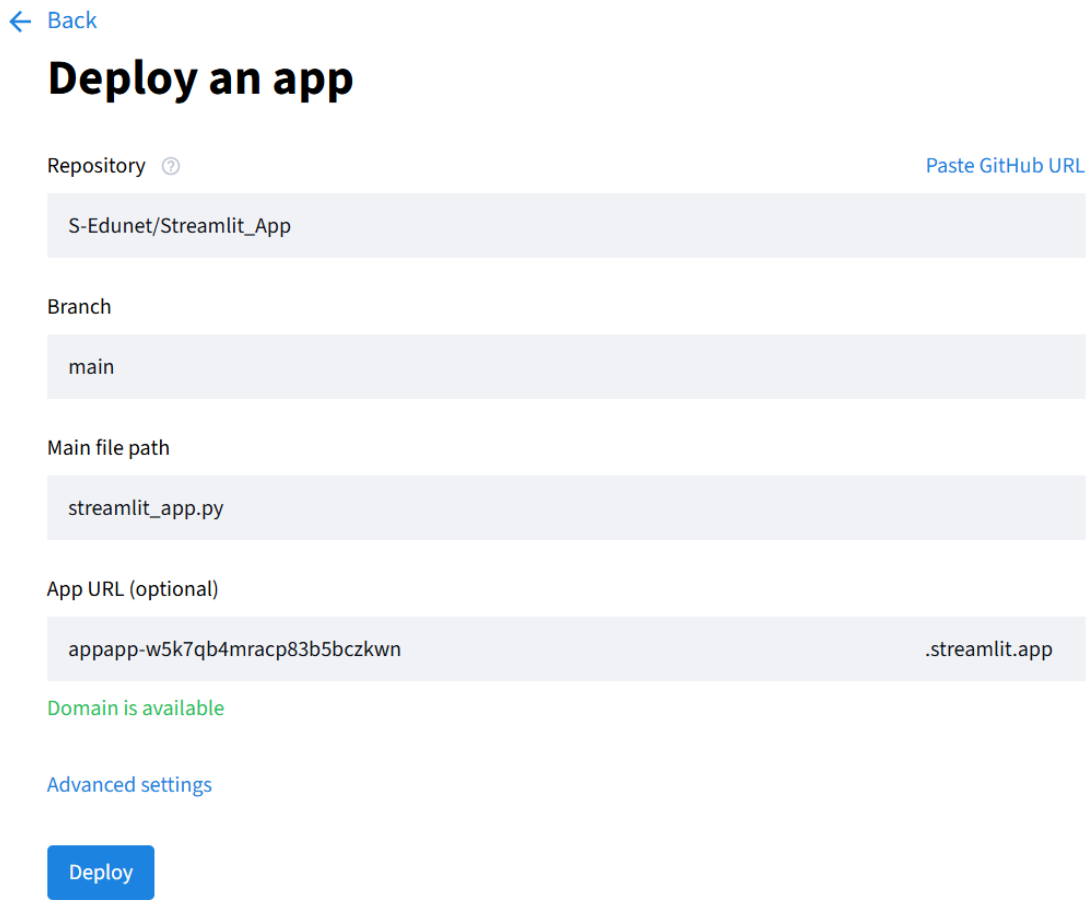


2. **Click on Create App option (Top Right)**

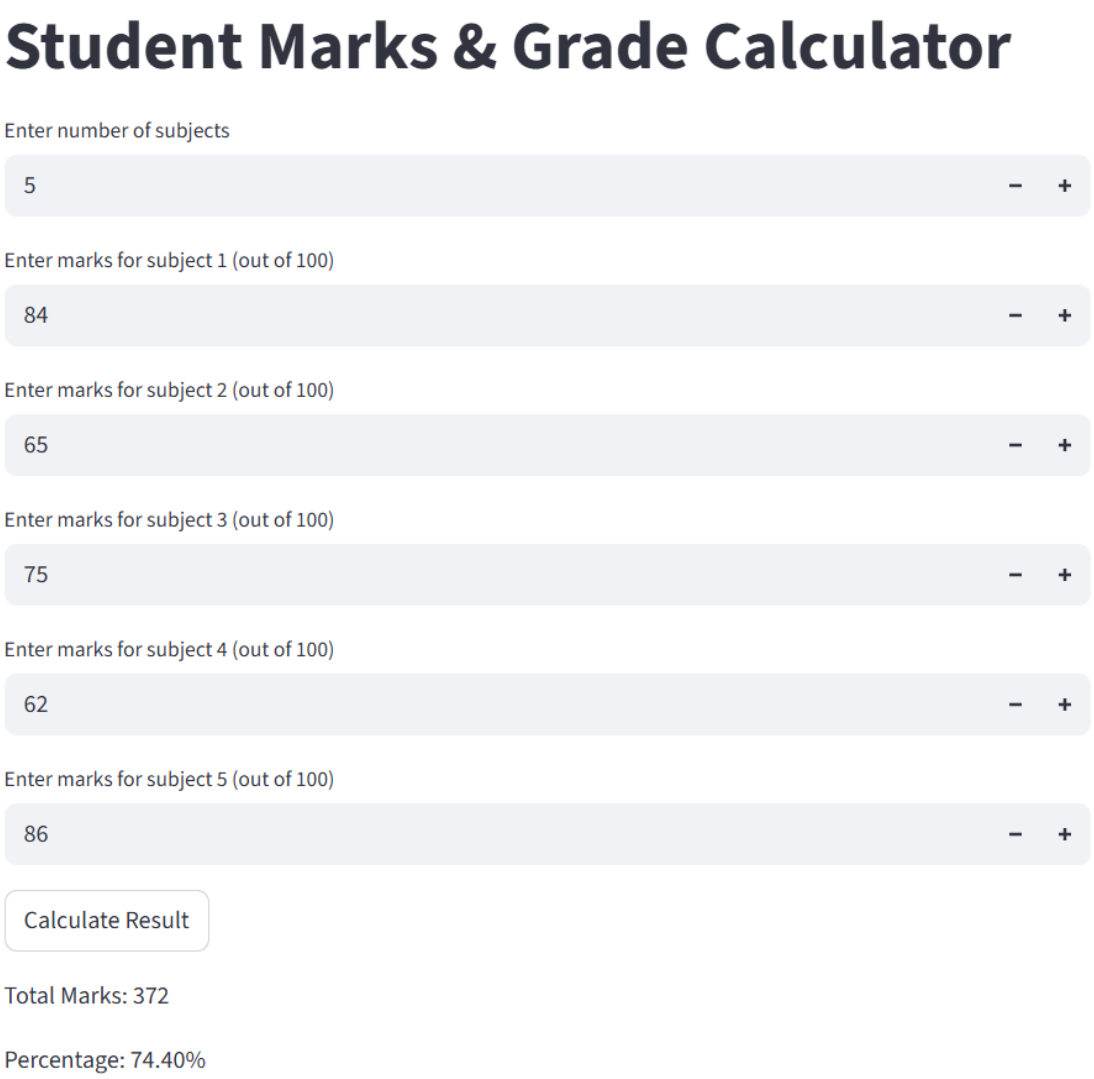
3. **Now select 'Deploy a public app from GitHub'**



4. **Select your Repository Name in which you have added the files. Click on Deploy.**



Output



Application 2: Placement Data Analyzer

1. Objective

To create an application that allows faculty or students to upload placement data (CSV file) and instantly view data summaries and CTC distribution using interactive charts.

2. Equipment Required

- A computer or laptop with internet connectivity
- Python (3.9 or above) installed
- VS Code or any code editor
- Web browser (Google Chrome/Edge)
- Sample CSV file with placement data
- GitHub account
- Streamlit Community Cloud account

3. Prerequisites

- Basic knowledge of Python programming (variables, loops, conditionals)
- Familiarity with using pip to install Python libraries: streamlit and pandas
- Basic understanding of GitHub usage (optional for deployment stage)

4. Problem Statement

Analyzing placement data manually using spreadsheets can be inefficient. This project aims to automate the analysis by enabling users to upload a CSV file and view placement data distribution through a simple web interface.

5. Procedure

a. Create a new folder.
b. Create a Python file pacement_data.py with code to:
    - Upload CSV file.
    - Display dataset in a table.
    - Show a bar chart for CTC distribution.
c. Create a requirements.txt file with: streamlit and pandas
d. Push the project to GitHub.
e. Deploy the app on Streamlit Community Cloud.

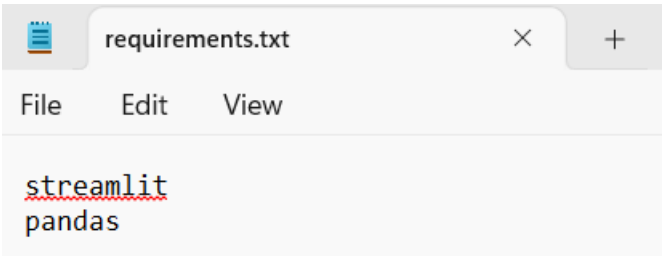
Steps:

Perform the following steps on the local machine

1. Create a new folder. We will add the python code file and the requirements file in this folder.
2. Create a new python file. Here we are calling it placement_data.py

```
import streamlit as st
import pandas as pd

st.title("Placement Data Analyzer")

uploaded_file = st.file_uploader("Upload placement CSV file", type="csv")

if uploaded_file is not None:
    df = pd.read_csv(uploaded_file)
    st.write("Dataset Preview:")
    st.dataframe(df)

    if "CTC" in df.columns:
        st.subheader("CTC Distribution")
        st.bar_chart(df["CTC"])
    else:
        st.warning("Please ensure there is a 'CTC' column in the dataset.")
else:
    st.info("Upload a CSV file to analyze placement data.")
```

3. Create a new requirements.txt file.



4. **Follow the steps of GitHub and Streamlit as shown in Application 1.**

Output

Placement Data Analyzer

Upload placement CSV file

Drag and drop file here
Limit 200MB per file • CSV

Browse files

placement_data.csv 5.0KB

Dataset Preview:

	Name	Branch	CTC	Final_Year_Marks(%)
0	Priya Sharma	EEE	7	86.8
1	Ankit Reddy	ME	3.3	51.3
2	Sneha Patel	ECE	6	86.1
3	Priya Mehta	ME	5.3	75.3
4	Priya Gupta	ME	4.2	66.3
5	Kiran Sharma	IT	6.4	64.5
6	Amit Verma	ECE	8.6	80.4
7	Sneha Reddy	ECE	8.5	72.3
8	Neha Gupta	ECE	6.3	54.7
9	Amit Mehta	ME	5.4	73.6

Application 3: Basic Coding Quiz App

1. **Objective**

   To create a simple coding quiz application where students can attempt multiple-choice questions and see immediate feedback and final scores.

2. **Equipment Required**

   - Laptop or desktop with Python installed
   - Code editor (VS Code)
   - Web browser
   - GitHub and Streamlit accounts

3. **Prerequisites**

   - Basic knowledge of Python programming (variables, loops, conditionals)
   - Familiarity with using pip to install Python libraries
   - Basic understanding of how to run a Python script from the terminal
   - Basic understanding of GitHub usage (optional for deployment stage)

4. **Problem Statement**

This app aims to provide a platform where students can attempt a small quiz online and view their scores in real time.

5. **Procedure**

   a. Create a new folder.

   b. Create a Python file quiz_app.py with code to:

      - Display quiz questions.
      - Accept answers using radio buttons.
      - Show correct/incorrect feedback.
      - Display final score.

   c. Create a requirements.txt file containing: streamlit

   d. Push the app to a GitHub repository.

   e. Deploy using Streamlit Community Cloud.

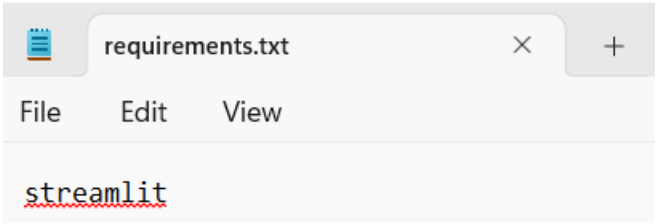
Steps:

Perform the following steps on the local machine

1. Create a new folder. We will add the python code file and the requirements file in this folder.
2. Create a new python file. Here we are calling it quiz_app.py

```
import streamlit as st

st.title("Coding Quiz")

questions = [
    {"q":"Which language is primarily used for AI/ML?",
"options":["C++","Python","HTML"], "answer":"Python"},
    {"q":"What does HTML stand for?", "options":["Hyperlinks and Text Markup
Language","Hyper Text Markup Language","Home Tool Markup Language"],
"answer":"Hyper Text Markup Language"}
]

if "score" not in st.session_state:
    st.session_state["score"] = 0
if "submitted" not in st.session_state:
    st.session_state["submitted"] = [False] * len(questions)

for i, ques in enumerate(questions):
    ans = st.radio(ques["q"], ques["options"], key=i)
    if st.button(f"Submit Q{i+1}", key=f"btn{i}"):
        if not st.session_state["submitted"][i]:
            if ans == ques["answer"]:
                st.success("Correct!")
                st.session_state["score"] += 1
            else:
                st.error(f"Wrong! Correct answer: {ques['answer']}")
            st.session_state["submitted"][i] = True

if all(st.session_state["submitted"]):
    st.write(f"Your final score is {st.session_state['score']} out of
{len(questions)}")
```

3. Create a new requirements.txt file.



4. **Follow the steps of GitHub and Streamlit as shown in Application 1.**

Output