## a. Title Page

**Title: Student Dropout Risk Prediction Using Machine Learning**

**Name: Lucky Singh**

**Roll No.: 202401100300146**

**Branch: CSE (AI)**

**Subject: Artificial Intelligence**

**Section:B**

**KIET Group of Institutions**

**Date: 22/04/2025**

## b. Introduction

The dropout rate among students has been a major concern in the education sector. Identifying at-risk students at an early stage can help institutions implement remedial measures to support them. This project aims to develop a predictive model using machine learning to identify students who are at risk of dropping out based on their academic performance and classroom behavior. The model uses features like **attendance**, **grades**, and **participation** to predict the **dropout_risk**.

## c. Methodology

1. **Dataset Loading:** The dataset student_dropout_data.csv was loaded using pandas.
2. **Data Preprocessing:**
   - The target column dropout_risk was converted to binary (yes → 1, no → 0).
   - Selected input features: attendance, grades, participation.
3. **Data Splitting:** The dataset was split into training (80%) and testing (20%) using train_test_split.

4. **Model Building:** A **Decision Tree Classifier** from sklearn was used to train on the dataset.
5. **Evaluation:** The model's performance was evaluated using Accuracy Score, Confusion Matrix, and Classification Report.
6. **User Input:** The model accepts user input from the console to predict the dropout risk for a given student.

---

**d. Code**

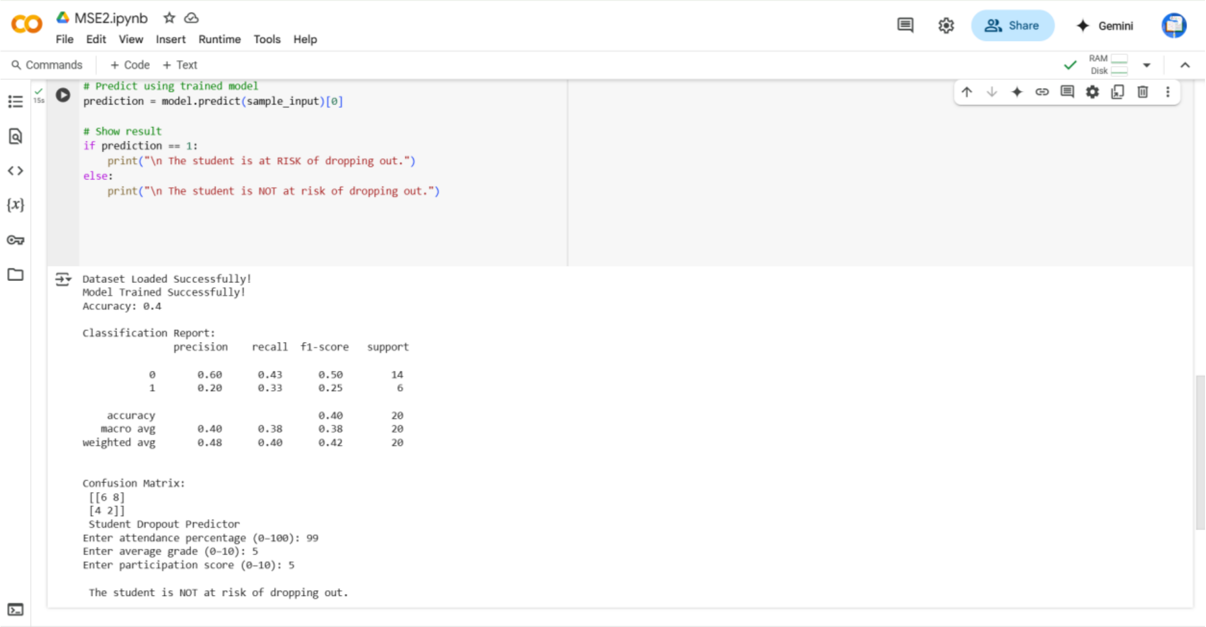
python

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import numpy as np

# Load dataset
df = pd.read_csv("student_dropout_data.csv")
print("Dataset Loaded Successfully!")
df.head()

# Convert target labels to numeric: 'yes' -> 1, 'no' -> 0
df['dropout_risk'] = df['dropout_risk'].map({'yes': 1, 'no': 0})

# Features and Target
X = df[['attendance', 'grades', 'participation']]
y = df['dropout_risk']
```

```
# Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Train model
model = DecisionTreeClassifier()
model.fit(X_train, y_train)
print("Model Trained Successfully!")

# Predict and evaluate
y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))

# User input
print(" Student Dropout Predictor")
attendance = float(input("Enter attendance percentage (0–100): "))
grades = float(input("Enter average grade (0–10): "))
participation = int(input("Enter participation score (0–10): "))

# Predict for sample input
sample_input = pd.DataFrame(np.array([[attendance, grades, participation]]),
                        columns=["attendance", "grades", "participation"])
prediction = model.predict(sample_input)[0]
```

```
# Output result
if prediction == 1:
    print("\n The student is at RISK of dropping out.")
else:
    print("\n The student is NOT at risk of dropping out.")
```

## e. Output/Result



```
# Predict using trained model
prediction = model.predict(sample_input)[0]

# Show result
if prediction == 1:
    print("\n The student is at RISK of dropping out.")
else:
    print("\n The student is NOT at risk of dropping out.")
```

```
Dataset Loaded Successfully!
Model Trained Successfully!
Accuracy: 0.4

Classification Report:
               precision    recall  f1-score   support

           0       0.60      0.43      0.50        14
           1       0.20      0.33      0.25         6

    accuracy                           0.40        20
   macro avg       0.40      0.38      0.38        20
weighted avg       0.48      0.40      0.42        20


Confusion Matrix:
 [[6 8]
 [4 2]]
 Student Dropout Predictor
Enter attendance percentage (0-100): 99
Enter average grade (0-10): 5
Enter participation score (0-10): 5

 The student is NOT at risk of dropping out.
```

## f. References/Credits

- Dataset: student_dropout_data.csv (source: [Your Data Source, if from Kaggle/UCI/etc.])
- Python Libraries:
    - pandas Documentation
- Code Implementation: Developed by **Lucky Singh**