AI CODE EXPLAINER

NAME-PREMANANDA DASH

REG NO-23BCE0950

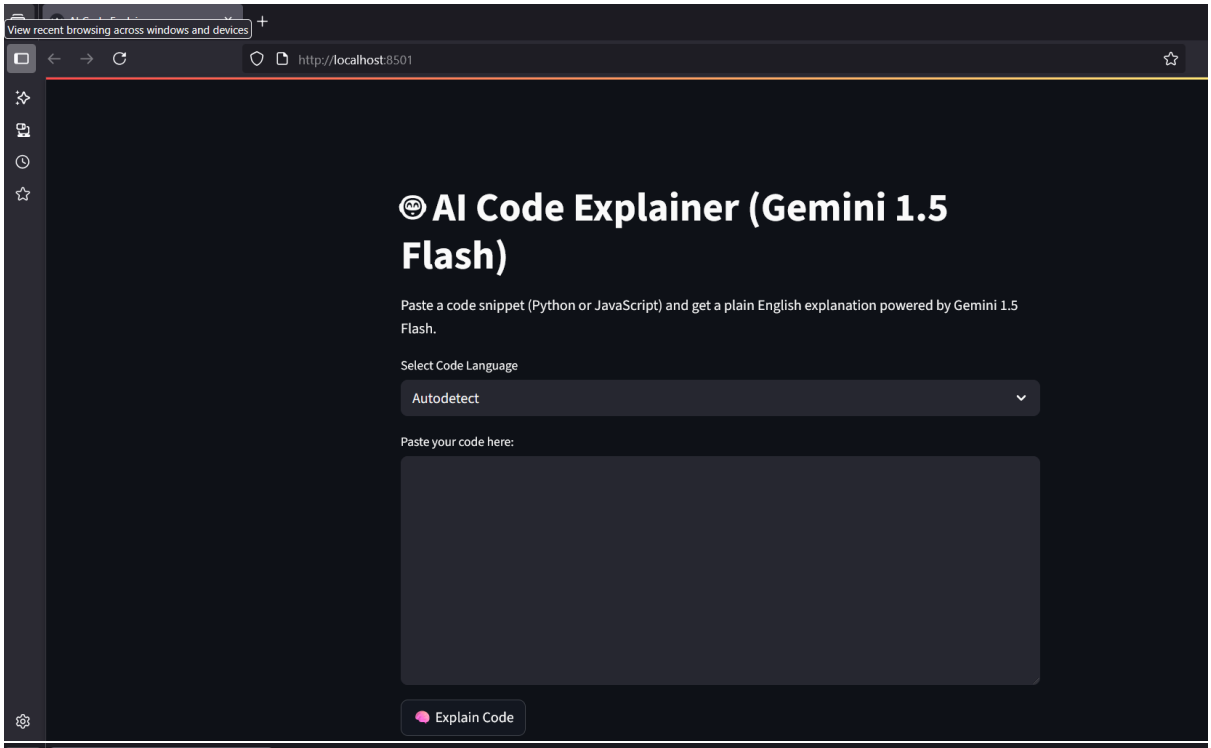
Vit Vellore University

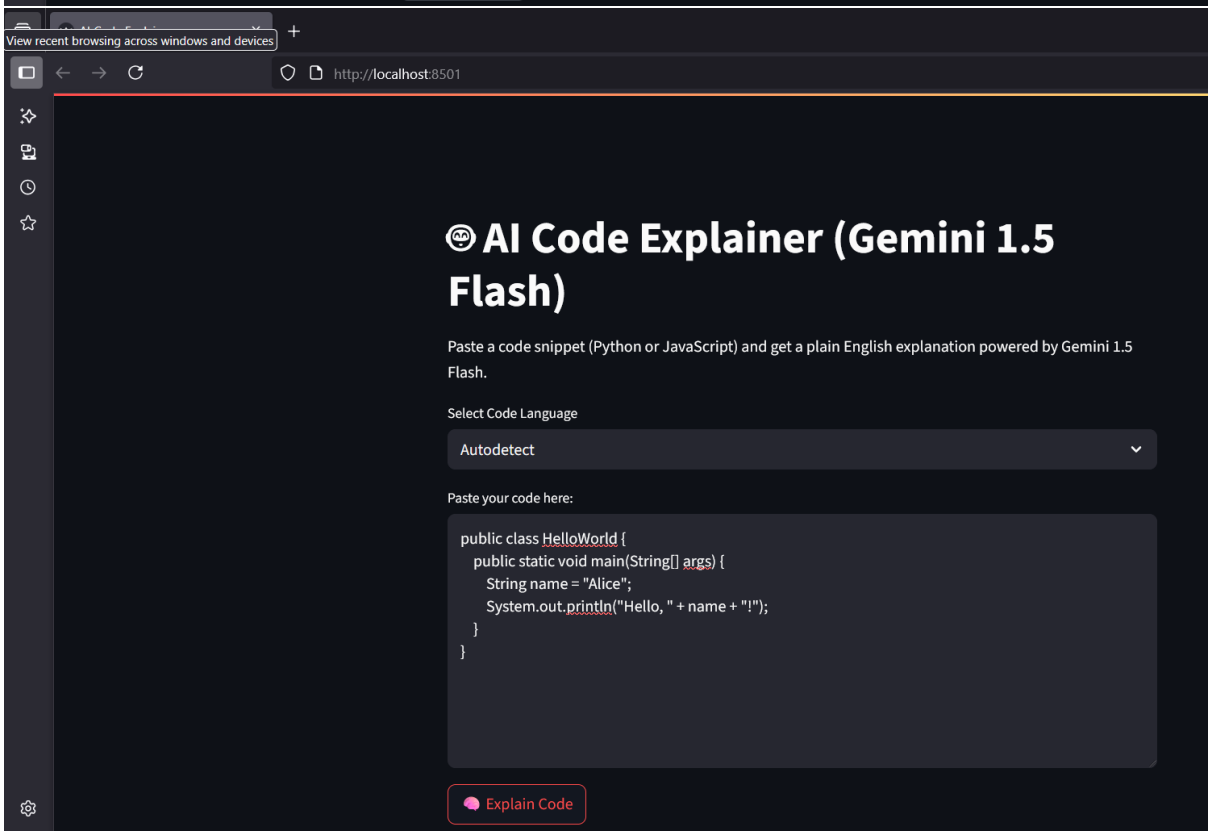
CODE-:

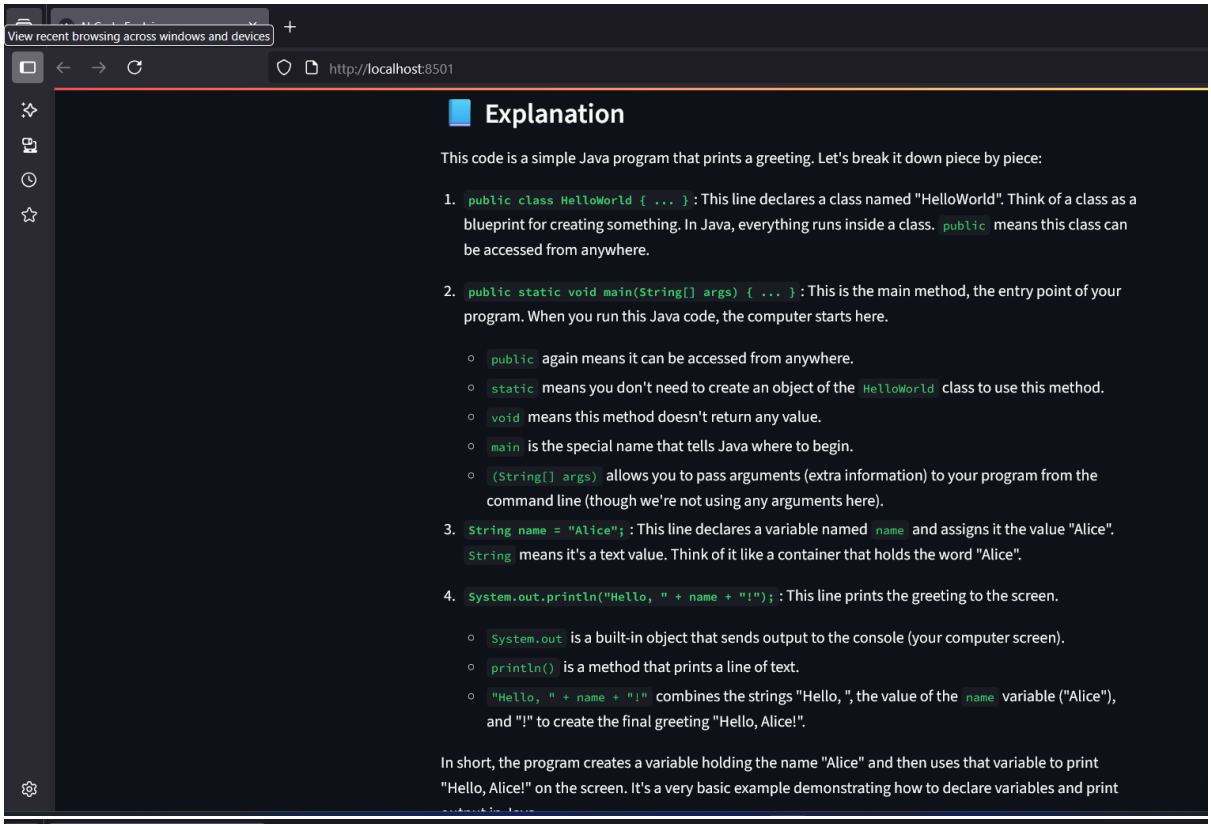
app.py > ...

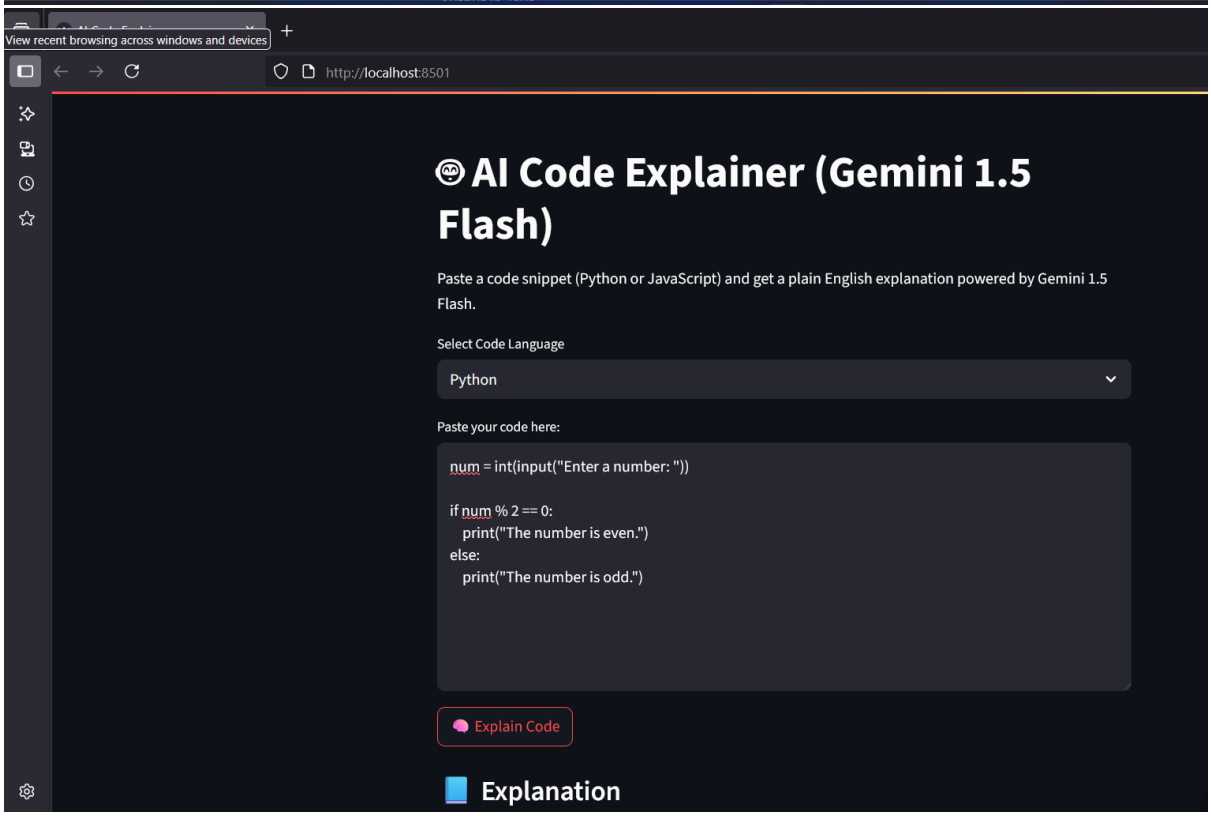
```python
import streamlit as st
import google.generativeai as genai

# Set your Gemini API key directly
GEMINI_API_KEY = "AIzaSyC755AH8TnQvNM7mksjazWfeEqKurdUtsU"
genai.configure(api_key=GEMINI_API_KEY)

# Initialize Gemini model
model = genai.GenerativeModel("gemini-1.5-flash")

# Streamlit UI setup
st.set_page_config(page_title="AI Code Explainer", layout="centered")
st.title("🤖 AI Code Explainer (Gemini 1.5 Flash)")

st.write("Paste a code snippet (Python or JavaScript) and get a plain English explanation powered by Gemini 1.5 Flash.")

# Language selection
language = st.selectbox("Select Code Language", ["Autodetect","Python", "JavaScript","C","kotlin","Java"])

# Code input area
code_input = st.text_area("Paste your code here:", height=250)

# Explain button
if st.button("🧠 Explain Code"):
    if not code_input.strip():
        st.warning("Please enter some code to explain.")
    else:
        with st.spinner("Explaining the code using Gemini..."):
            prompt = f"Explain the following {language} code in simple, beginner-friendly English:\n\n{code_input}"
            try:
                response = model.generate_content(prompt)
                st.subheader("📘 Explanation")
                st.write(response.text)
            except Exception as e:
                st.error(f"Error: {e}")
```

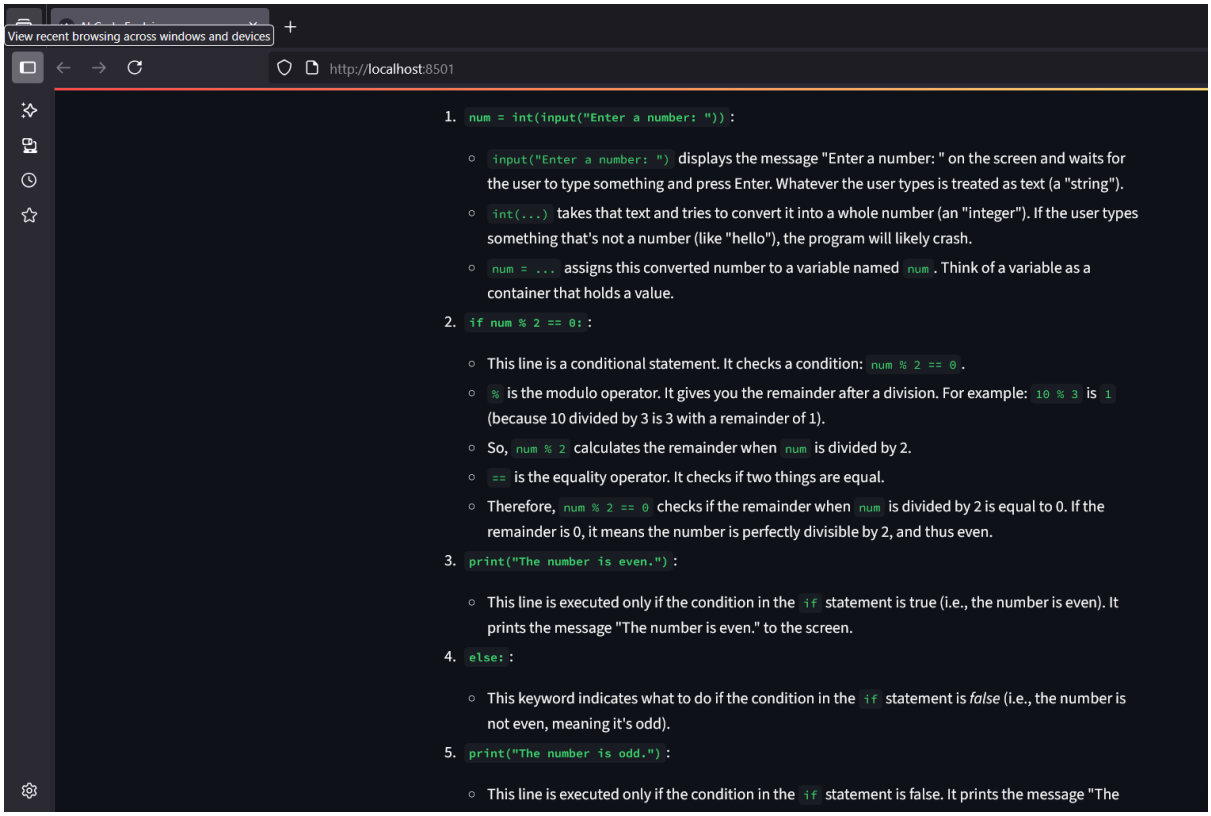
OUTPUT-:


View recent browsing across windows and devices
http://localhost:8501
AI Code Explainer (Gemini 1.5 Flash)
Paste a code snippet (Python or JavaScript) and get a plain English explanation powered by Gemini 1.5 Flash.
Select Code Language
Autodetect
Paste your code here:
Explain Code


View recent browsing across windows and devices
http://localhost:8501
AI Code Explainer (Gemini 1.5 Flash)
Paste a code snippet (Python or JavaScript) and get a plain English explanation powered by Gemini 1.5 Flash.
Select Code Language
Autodetect
Paste your code here:
public class HelloWorld {
public static void main(String[] args) {
String name = "Alice";
System.out.println("Hello, " + name + "!");
}
}
Explain Code


View recent browsing across windows and devices
http://localhost:8501
Explanation
This code is a simple Java program that prints a greeting. Let's break it down piece by piece:
1. public class HelloWorld { ... } : This line declares a class named "HelloWorld". Think of a class as a blueprint for creating something. In Java, everything runs inside a class. public means this class can be accessed from anywhere.
2. public static void main(String[] args) { ... } : This is the main method, the entry point of your program. When you run this Java code, the computer starts here.
public again means it can be accessed from anywhere.
static means you don't need to create an object of the HelloWorld class to use this method.
void means this method doesn't return any value.
main is the special name that tells Java where to begin.
(String[] args) allows you to pass arguments (extra information) to your program from the command line (though we're not using any arguments here).
3. String name = "Alice"; : This line declares a variable named name and assigns it the value "Alice". String means it's a text value. Think of it like a container that holds the word "Alice".
4. System.out.println("Hello, " + name + "!"); : This line prints the greeting to the screen.
System.out is a built-in object that sends output to the console (your computer screen).
println() is a method that prints a line of text.
"Hello, " + name + "!" combines the strings "Hello, ", the value of the name variable ("Alice"), and "!" to create the final greeting "Hello, Alice!".
In short, the program creates a variable holding the name "Alice" and then uses that variable to print "Hello, Alice!" on the screen. It's a very basic example demonstrating how to declare variables and print


View recent browsing across windows and devices
http://localhost:8501
AI Code Explainer (Gemini 1.5 Flash)
Paste a code snippet (Python or JavaScript) and get a plain English explanation powered by Gemini 1.5 Flash.
Select Code Language
Python
Paste your code here:
num = int(input("Enter a number: "))
if num % 2 == 0:
print("The number is even.")
else:
print("The number is odd.")
Explain Code
Explanation



1. `num = int(input("Enter a number: "))` :
   - `input("Enter a number: ")` displays the message "Enter a number: " on the screen and waits for the user to type something and press Enter. Whatever the user types is treated as text (a "string").
   - `int(...)` takes that text and tries to convert it into a whole number (an "integer"). If the user types something that's not a number (like "hello"), the program will likely crash.
   - `num = ...` assigns this converted number to a variable named `num`. Think of a variable as a container that holds a value.
2. `if num % 2 == 0:` :
   - This line is a conditional statement. It checks a condition: `num % 2 == 0`.
   - `%` is the modulo operator. It gives you the remainder after a division. For example: `10 % 3` is `1` (because 10 divided by 3 is 3 with a remainder of 1).
   - So, `num % 2` calculates the remainder when `num` is divided by 2.
   - `==` is the equality operator. It checks if two things are equal.
   - Therefore, `num % 2 == 0` checks if the remainder when `num` is divided by 2 is equal to 0. If the remainder is 0, it means the number is perfectly divisible by 2, and thus even.
3. `print("The number is even.")` :
   - This line is executed only if the condition in the `if` statement is true (i.e., the number is even). It prints the message "The number is even." to the screen.
4. `else:` :
   - This keyword indicates what to do if the condition in the `if` statement is *false* (i.e., the number is not even, meaning it's odd).
5. `print("The number is odd.")` :
   - This line is executed only if the condition in the `if` statement is false. It prints the message "The