# Chapter - 26 Introduction to AI Coding in python

**Lesson Objective :**

- How AI coding works Coding using functions
- Basic programming logics

**Skills to be attained :** Knowledge on AI coding

**Tools / Websites / Resources :**

1. https://www.onlinegdb.com//
2. gedit text editor in Hitech lab

**Teacher Led Instructions :**

Artificial Intelligence (AI) coding is about teaching computers to learn from data, make decisions, and solve problems in ways similar to humans. Imagine training a computer to recognize pictures, suggest songs you might like, or even help predict the weather. Here's how AI coding works in simple steps:

**1. Learning from Data**

AI learns by studying data. For example, to help a computer recognize cats in photos, we show it many pictures labeled as "cat" or "not cat." This is called **training**. This type is called as Supervised machine learning.

**2. Finding Patterns**

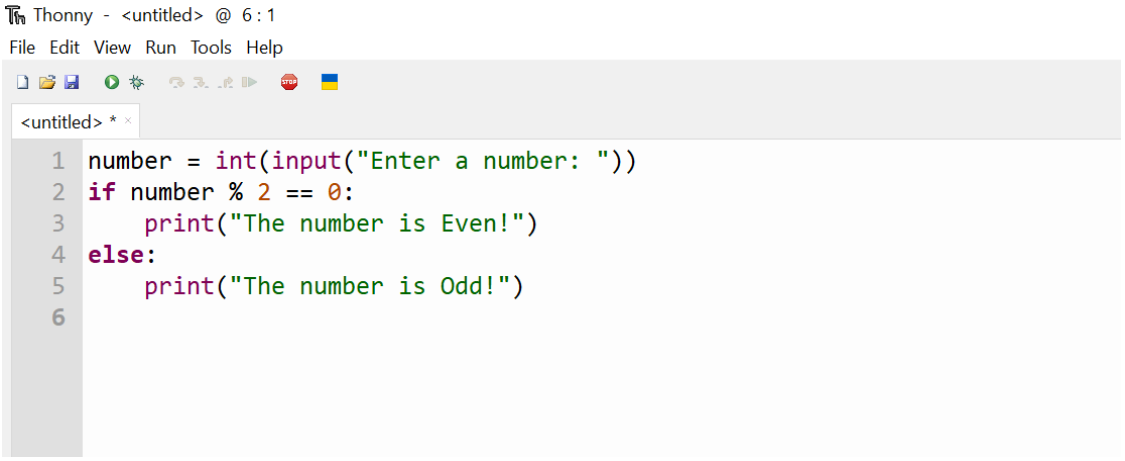
The computer looks for patterns in the data, such as what cats usually look like (ears, tails, whiskers). It then uses these patterns to make decisions. This is called Unsupervised **machine learning** because the computer learns from examples.

**3. Making Predictions**

Once it learns, the AI can make predictions on new data. For instance, it can now look at a new photo and tell you if it thinks it's a cat or not. This part is called **prediction**.

Here's a very basic example of AI code in Python, which detects if a number is odd or even. This can help students grasp the basics of programming logic before diving into AI concepts.

**1. Simple Odd or Even Program (to build logic)**



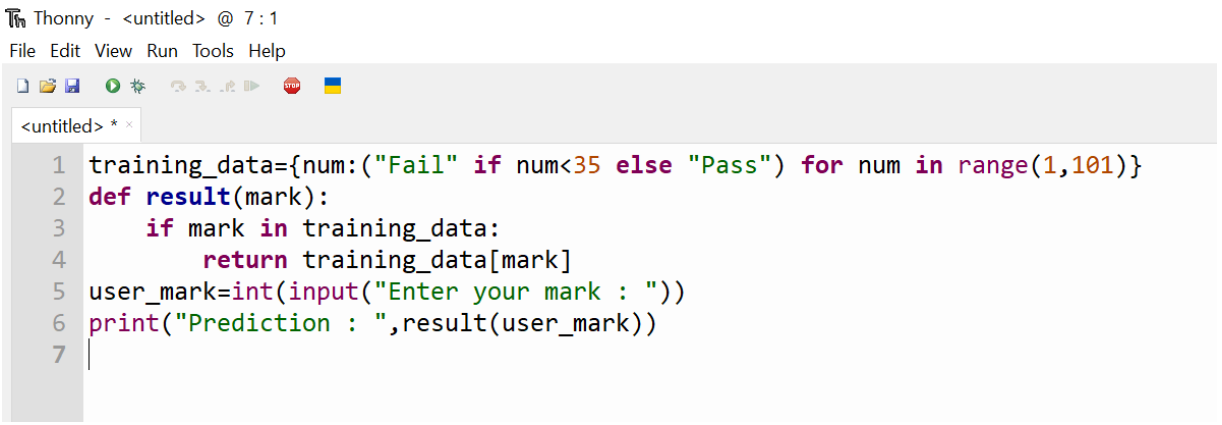

This example can introduce students to the idea of logical thining in AI with very minimal code.

print("The number is Even!") else: print("The number is Odd!")

**2. Basic AI-inspired Code: Number Guessing (Pattern Matching)**

In this example, the code "learns" a pattern and makes predictions based on it, similar to how simple machine learning models work.

We train the machine to learn the criteria for pass and fail. When you enter the mark, it will say whether you are pass or fail from the data it has.




**How It Works:**

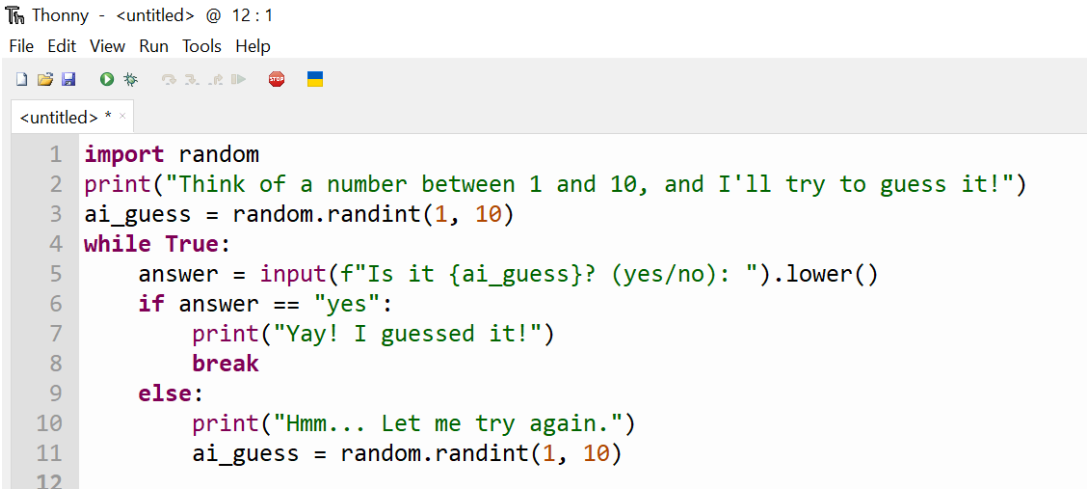
- The program "learns" from the training data dictionary.
- If it doesn’t recognize the number, it uses a simple rule (like a basic machine learning "model") to predict.

This example can introduce students to the idea of pattern recognition in AI with very minimal code.

Here's a very simple AI-inspired code in Python that introduces school students to a basic concept of AI: a "guessing game" using if-else logic and random choice. This doesn't dive deep into AI but lays a foundation by creating an interactive experience that mimics AI guessing or making decisions.

### 3. Simple Number Guessing Game

In this example, the AI will "try" to guess a number the student is thinking of by choosing a random number and checking if it's correct.




```python
import random
print("Think of a number between 1 and 10, and I'll try to guess it!")
ai_guess = random.randint(1, 10)
while True:
    answer = input(f"Is it {ai_guess}? (yes/no): ").lower()
    if answer == "yes":
        print("Yay! I guessed it!")
        break
    else:
        print("Hmm... Let me try again.")
        ai_guess = random.randint(1, 10)
```

# AI makes a new random guess

**Explanation of AI Concepts in the Code:**

1. **Random Guessing**: The AI starts with no knowledge and "learns" by randomly guessing. This teaches that early AI systems don't always start with much "intelligence."

2. **Feedback Loop**: The AI guesses until it receives feedback that it's correct. This introduces the idea of feedback-based learning, a basic concept in AI.

3. **Iteration for Improvement**: Although the AI here doesn't retain memory, this loop mimics how real AI systems can "try" repeatedly to improve outcomes

Activities :

1. Write a python code to check whether the given number is positive or negative.

2. Writer a python code to Check a person is eligible to vote or not.

3. Write a code to check whether given number is divisible by 2 or not. Print “It is divisible” else “It is not divisible”.

**Conclusion :** This code provides a hands-on way for students to interact with simple decision-making and random choices, helping them understand the basics of guessing, learning, and iteration. They can begin experimenting with more advanced patterns and logic writing code for similar concepts.