

# **Maharaja Institute of Technology Mysore.**

AN AUTONOMOUS INSTITUTE AFFILIATED TO VTU.



A REPORT ON

**The trends and Innovations in AI**

Submitted for additional requirement for earning 4 years Degree Program for

**BACHELOR OF ENGINEERING**

**IN**

**CSE-ARTIFICIAL INTELLIGENCE**

**UNDER THE GUIDENCE OF**

**PROF Dr. Agughasi Victor I.**



**Department of CSE-AI**

**MIT MYSORE 2024-2025**

# ACKNOWLEDGEMENT

Before we turn to the AICTE activity program work ,we would like to add a few heart felt words for the people who have been part of this work by supporting and encouraging us.

At the outset ,we would like to thank almighty for making us as we are .We would like take this opportunity to express my honour , respect , deepest gratitude , and genuine regards to our beloved principal [DR . NARESH KUMAR B. G.](#), Maharaja Institute of Technology Mysore , for giving us all the guidance required for our work apart being a constant source of inspiration. We express our sincere thanks to [Dr. SHIVAMUTRYR. C.](#), PROFESSOR & HOD , Department of CSE, for his continuous encouragement ,guidance and moral support throughout the AICTE Activity program work.

We express our sincere thanks to our guide [Dr. Agughasi Victor I.](#), Mini Project and Technical Seminar Coordinator, Department of CSE (AI).for his guidance, constant, value able suggestions and encouragement Provided throughout the project work.

## LIST OF GROUP OF STUDENTS ARE LISTED BELOW

Sl.No.	USN	Name	Signature
1	4MH23CA051	Shreyas A	
2	4MH23CA058	Thejas A N	

## ABSTRACT

Artificial Intelligence (AI) is experiencing transformative growth, reshaping industries and societal norms. Key trends include the rise of generative AI, which empowers users to create realistic images, text, and multimedia content, thereby enhancing creativity and personalization. Machine learning models are becoming increasingly sophisticated, with advancements in natural language processing enabling more intuitive human-computer interactions.

Ethics in AI is gaining traction as organizations seek to address issues of bias, transparency, and accountability in algorithmic decision-making. The integration of AI with Internet of Things (IoT) technologies is enhancing data-driven insights, fostering smarter cities and more efficient resource management. Additionally, AI applications in healthcare are revolutionizing diagnostics, treatment planning, and patient care.

As AI continues to advance, its implications for workforce dynamics and economic structures are profound, necessitating ongoing dialogue about its role in society and the future of work. Overall, the trends in AI signal a shift towards more intelligent, adaptive systems that can enhance human capabilities while also raising important ethical considerations.

# MAHARAJAINSTITUTEOFTECHNOLOGYMYSORE

## DEPARTMENTOFCOMPUTERSCIENCE&ENGINEERING



# CERTIFICATE



*Certified at the mini-project worked on titled “Trend and Innovation of AI” is a beneficiary work carried out by **Shreyas A** (4MH23CA051), **Thejas A N** (4MH23CA058), in the partial fulfillment for the award of degree of Bachelor of Engineering in **CSE-AI** of the Visvesvaraya Technological University, Belagavi during the academic year 2024-25. It is certified that all corrections/suggestions indicated have been incorporated in the report. The project report has been approved as it satisfies the academic requirements with respect to the Project work prescribed for Bachelor of Engineering Degree.*

Signature of the Guide

Dr. Agughasi Victor I.,

Mini Project and Technical  
Seminar Coordinator,  
Department of CSE (AI).

Signature of the HOD

Dr. Shivamurthy R C

Professor & Head  
Dept. of CSE

Signature of the Principal

Dr. B G Naresh Kumar

Principal  
MIT Mysore

# INTRODUCTION:

## Define artificial intelligence ?

Artificial intelligence (AI) refers to the development of computer systems or machines that can perform tasks requiring human-like intelligence. These tasks include understanding language, recognizing patterns, solving problems, learning from experience, and making decisions. AI encompasses technologies like machine learning, natural language processing, and robotics, which enable machines to adapt and improve their performance over time without direct human intervention.

**John McCarthy**, an American computer scientist. He coined the term "Artificial Intelligence" in 1956 and was one of the organizers of the Dartmouth Conference that year, which is considered the founding event of AI as a field of study.

## Artificial Intelligence (AI) how it works & its applications:

- **Data Collection** : AI systems learn from vast amounts of data. This data can come from various sources, such as text, images, videos, or sensor readings.
- **Algorithms** : AI uses algorithms, which are sets of rules or instructions that guide the machine in processing data and making decisions.
- **Machine Learning** : This is a subset of AI where machines learn from data without being explicitly programmed.
- **Training Models** : During training, the AI system is fed data, and its model adjusts itself to make accurate predictions or decisions.
- **Neural Networks** : In more advanced AI systems, particularly deep learning, artificial neural networks are used.
- **Inference** : After training, the AI system is used to make predictions or decisions based on new, unseen data.
- **Feedback Loop** : AI systems often incorporate feedback mechanisms to refine their performance continuously.

## Applications:

- **Healthcare** : AI-powered diagnostic tools can analyze medical images (X-rays, MRIs) for early detection of diseases.
- **Finance** : Fraud detection algorithms can identify unusual transactions and prevent financial crimes.

- **Automotive** : AI assists with advanced driver-assistance systems (ADAS), such as collision detection, automatic braking, and lane-keeping.
- **Application Development:** AI can automate repetitive coding tasks, and help ensure code consistency.

## Latest Technologies Based on AI :

There are variety of inventions done in AI. Such



Tesla's Autopilot(ADAS)



Digital fingerprint concept



Apple vision pro



Alexa



ChatGPT

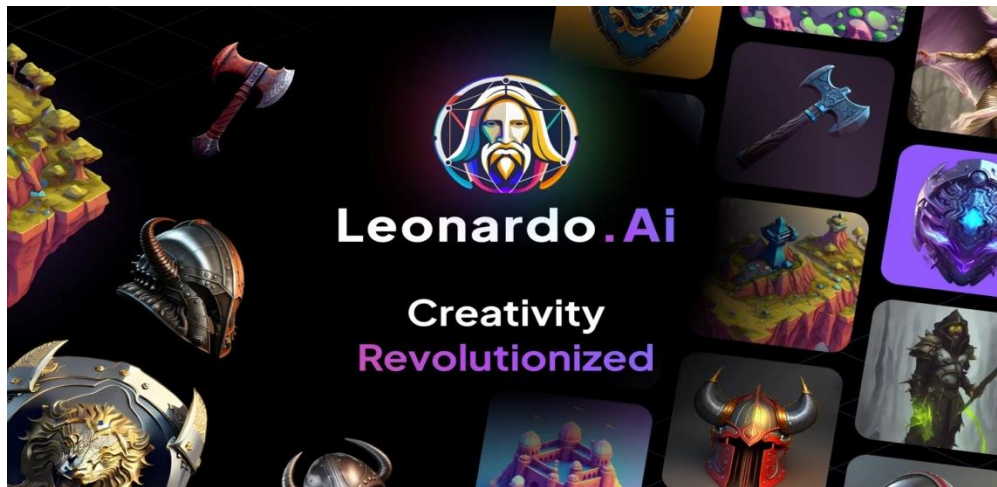


AI bots

## Here are some experiments with AI tools:

### How the [Leonardo.ai](#) works in situation of converting txt to image:

Leonardo AI, like other *\*text-to-image models\** such as OpenAI's DALL-E, works by using advanced machine learning techniques to generate images based on textual descriptions. The core of how these models function lies in their ability to *\*understand language and generate visuals\** that align with what's described. Below is a breakdown of how this process generally works:

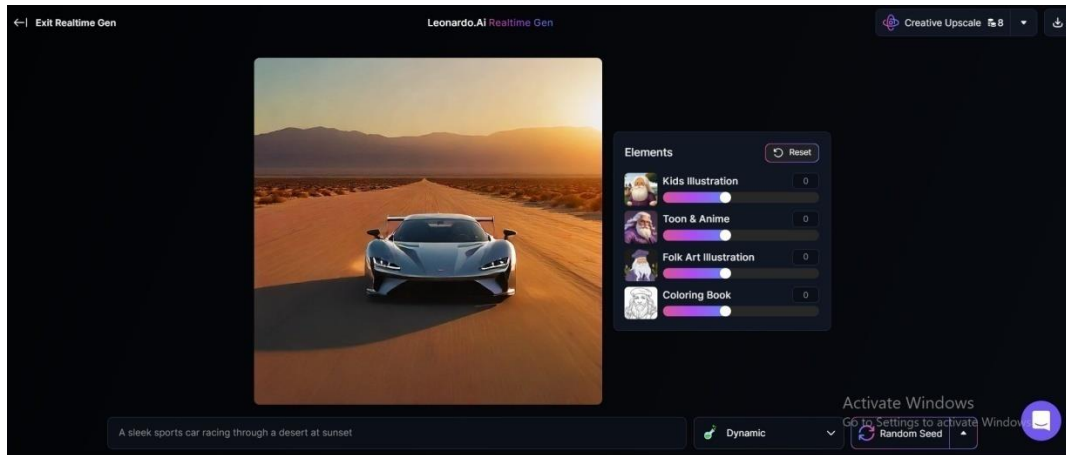


## 1. Understanding the Text Input:

The first step for Leonardo AI in converting text to an image is *\*natural language processing (NLP)*. This involves interpreting the input prompt to understand the *\*\*key elements, objects, and styles\** described.

**For example**, if the input is "A sleek sports car racing through a desert at sunset," the model identifies objects like "sports car," "desert," and "sunset" and understands their relationships and characteristics.





## 2. Text Embedding:

The text input is converted into a **numerical representation** using a technique called **text embedding**. This step transforms the descriptive words into a form that the AI can process. For instance, a phrase like "roaring engine" or "sleek car" is embedded into vectors that represent their semantic meaning. These embeddings help the model understand the **concepts** and **visual details** associated with these terms.

## 3. Image Generation via Diffusion or GAN Models:

Once the text is understood and embedded, Leonardo AI uses **image generation techniques** to turn the description into a visual.

**Diffusion Models:** Modern text-to-image models like **Leonardo** and **DALL-E** often use **diffusion models**. These models start with random noise and iteratively refine this noise into an image, guided by the information extracted from the text.

## 4. Fine-Tuning the Output:

Leonardo AI likely uses **multiple layers of refinement** to improve the output image based on the prompt's complexity.

## 5. Post-Processing and Output:

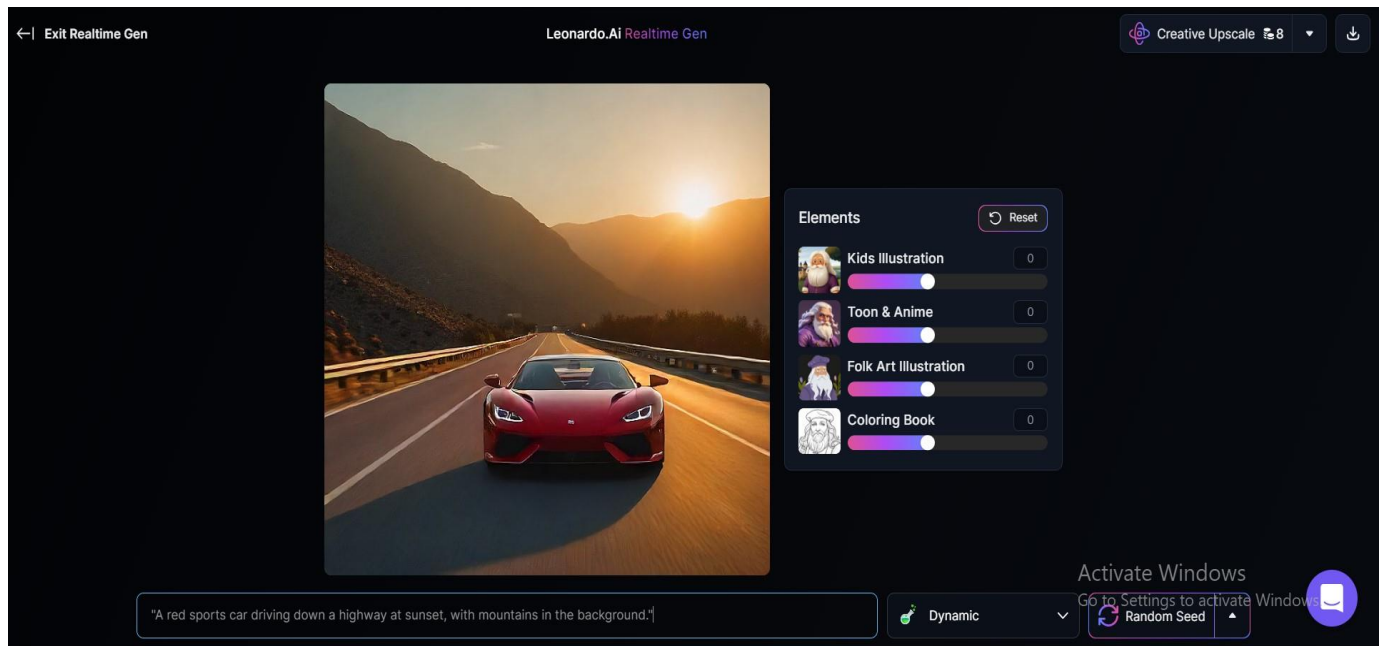
After generating the image, Leonardo AI performs '**post-processing**' to enhance the image quality, resolution, and details. This includes techniques like:

- **Super-resolution:** Upscaling the generated image to improve clarity and sharpness.
- **Detail Refinement:** Enhancing fine details such as textures or small objects based on the specific elements in the text.

Finally, the AI outputs the generated image, matching the text prompt as closely as possible.

## Example 01:-

"A red sports car driving down a highway at sunset, with mountains in the background."



Here is a Python code & process using the Leonardo AI:

```
# Make sure to install the required libraries
# pip install torch torchvision torchaudio transformers diffusers

import torch
from diffusers import StableDiffusionPipeline

# Load the pre-trained Stable Diffusion model
model_id = "CompVis/stable-diffusion-v1-4" # You may choose a different model
pipe = StableDiffusionPipeline.from_pretrained(model_id).to("cuda" if torch.cuda.is_available() else "cpu")

# Define your text prompt
text_prompt = "A red sports car driving down with the mountains in the background"

# Generate the image
with torch.no_grad():
    image = pipe(text_prompt).images[0]

# Save the image
image.save("red_sports_car.png")
```

- 1. Text Analysis :** The AI analyzes the prompt and identifies key elements: "red sports car," "highway," "sunset," "mountains."
- 2. Text Embedding :** The descriptive terms are converted into vectors that represent each element (e.g., what a "sports car" looks like, the color red, the highway, sunset lighting).

**3. Initial Noise:** The model begins with random noise and starts denoising it step by step, guided by the text embedding.

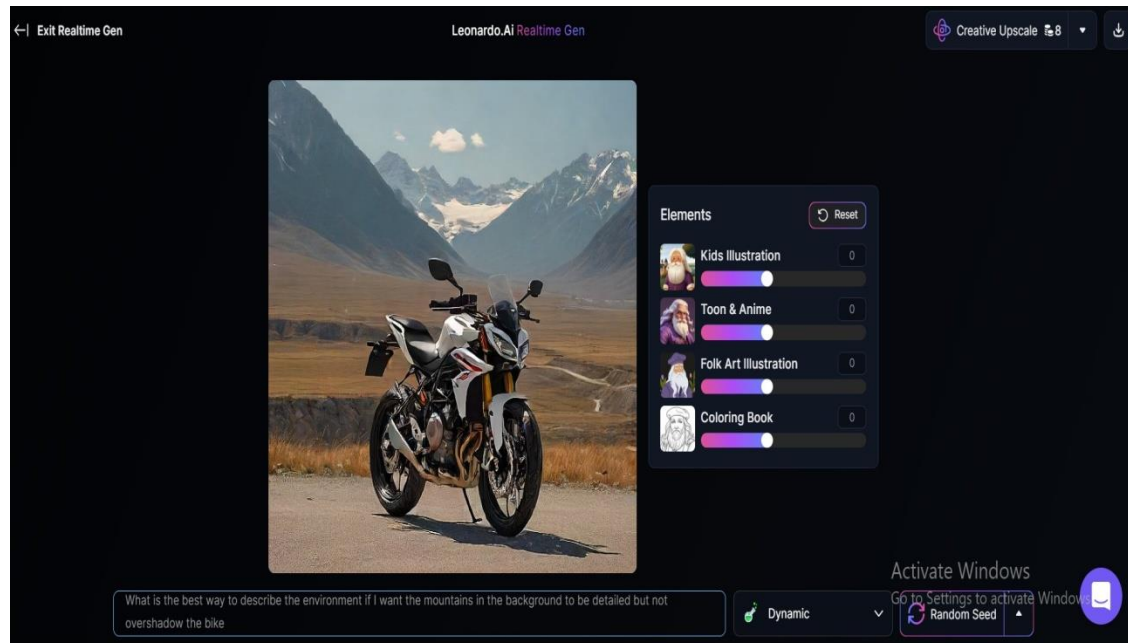
**4. Generation Steps:** The AI iterates through several steps:

- Adding the car's shape, the color red, the highway's layout, sunset lighting, and mountains in the background.
- Refining the image's details, such as the texture of the car, shadows from the sunset, and the mountains' colors.

**5. Final Output:** The result is a high-quality image of a red sports car driving on a highway at sunset with mountains in the background.

## Example 02:-

What is the best way to describe the environment if I want the mountains in the background to be detailed but not overshadow the bike?



Here is a Python code & process using the Leonardo AI:

```
import requests

# Define your API endpoint and key
API_URL = "https://api.leonardo.ai/generate" # Update with the actual endpoint
API_KEY = "YOUR_API_KEY"

# Define your text description
description = (
    "A scenic landscape featuring a bike in the foreground, "
    "positioned slightly off-center. In the background, "
    "majestic mountains with intricate textures and varying shades, "
    "muted colors to ensure the bike remains the focal point of the image."
)

# Set up the request payload
payload = {
    "prompt": description,
    "num_images": 1, # Number of images to generate
    "size": "1024x1024" # Specify the image size
}

# Set up the headers
headers = {
    "Authorization": f"Bearer {API_KEY}",
    "Content-Type": "application/json"
}

# Make the request to the API
response = requests.post(API_URL, headers=headers, json=payload)

# Check if the request was successful
if response.status_code == 200:
    image_data = response.json()
    image_url = image_data['images'][0] # Adjust based on API response structure
    print(f"Image generated successfully: {image_url}")
else:
    print(f"Error: {response.status_code}, {response.text}")
```

### **1. Start with the Scene Overview:**

Briefly introduce the overall setting of the scene. This helps the AI understand the type of environment you're envisioning.

### **2. Describe the Bicycle:**

Provide details of the bicycle first since it's the primary subject. Mention its position, size, and colors so it stands out from the rest of the scene.

### **3. Emphasize the Mountains:**

Mention that you want the mountains to be visible, detailed, and majestic but not overpowering the bike. Their placement should naturally frame the bike without distracting from it.

### **4. Define the Depth and Lighting:**

Talk about the lighting and how it can help separate the bike from the mountains.

### **5. Add Other Elements for Context (Optional):**

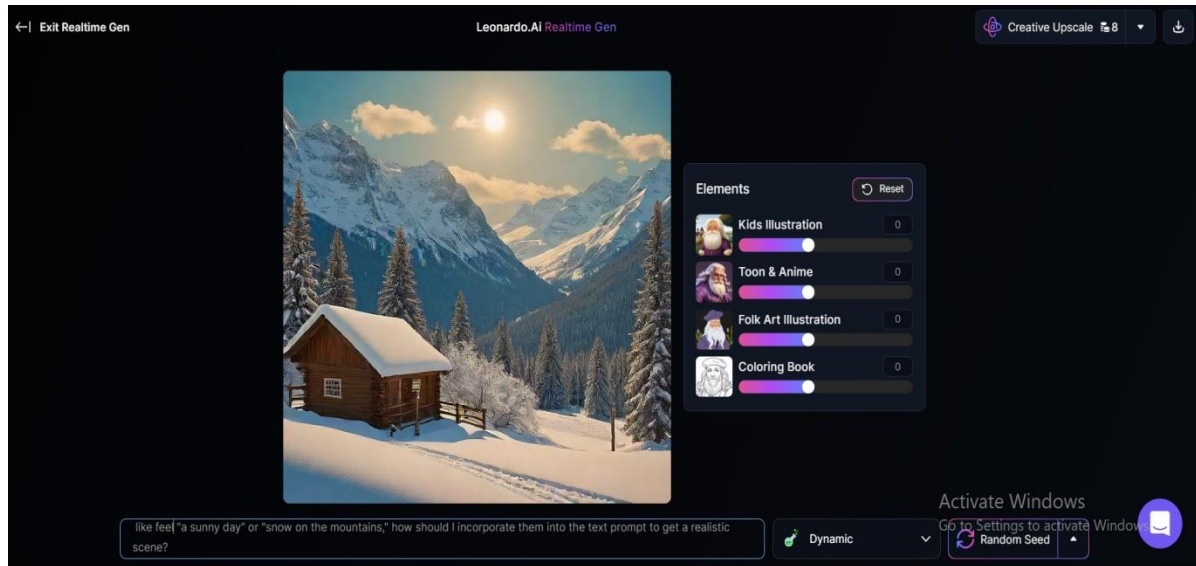
If you want more elements like trees, rocks, or pathways, add those to complete the scene.

### **6. Conclude with Focus Instructions:**

Reaffirm which element should draw the eye first.

## Example 03:-

Like feel "a sunny day" or "snow on the mountains," how should I incorporate them into the text prompt to get a realistic scene?



Here is a Python code & process using the Leonardo AI:

```
import requests

# Define your API endpoint and key
API_URL = "https://api.leonardo.ai/generate" # Replace with the actual endpoint
API_KEY = "YOUR_API_KEY" # Replace with your actual API key

# Define the text description for the image
description = (
    "A picturesque landscape featuring a bike in the foreground under a bright blue sky on a sunny day. "
    "The mountains in the background are detailed, with vibrant green hillsides and a few fluffy white clouds "
    "scattered across the sky, ensuring the bike remains the focal point."
)

# Set up the request payload
payload = {
    "prompt": description,
    "num_images": 1, # Number of images to generate
    "size": "1024x1024" # Specify the desired image size
}

# Set up the headers
headers = {
    "Authorization": f"Bearer {API_KEY}",
    "Content-Type": "application/json"
}

# Make the request to the API
response = requests.post(API_URL, headers=headers, json=payload)

# Check if the request was successful
if response.status_code == 200:
    image_data = response.json()
    image_url = image_data['images'][0] # Adjust based on API response structure
    print(f"Image generated successfully: {image_url}")
else:
    print(f"Error: {response.status_code}, {response.text}")
```

### 1. Establish the Setting (Time of Day, Weather Conditions):

- Define the overall mood, lighting, and weather.

## **2. Specify the Landscape (What is Visible?):**

- Describe the environment in detail, such as terrain, vegetation, water bodies, and any natural formations.
- **"A sunny day"**: Include rolling hills, a grassy field, trees, or a distant horizon.
- **"Snow on the mountains"**: Mention the shape and texture of the mountains, trees covered in snow, or icy rivers.

## **3. Describe Light and Shadow (Lighting Effects):**

- Explain how the light interacts with objects in the scene.
- **"A sunny day"**: Focus on the brightness and shadows cast by trees or hills, and the sun's warmth.
- **"Snow on the mountains"**: Discuss how sunlight reflects off the snow, casting cool, crisp shadows.

## **4. Add Atmospheric and Sensory Details (Temperature, Sound, and Feel):**

- Incorporate atmospheric details like breeze, temperature, or sounds.
- **"A sunny day"**: A gentle breeze, warm air, the scent of flowers, or birds chirping.
- **"Snow on the mountains"**: The crispness of the cold air, the crunch of snow underfoot, or the silence of the landscape.

## **5. Enhance with Color Descriptions:**

- Use colors to enrich the scene.
- **"A sunny day"**: Talk about the vibrant greens of grass, the blue of the sky, and the golden hues of sunlight.
- **"Snow on the mountains"**: Focus on the white snow, dark green trees, and the bluish tint of shadows.

## **6. Incorporate Human or Animal Elements (If Applicable)**

- If relevant, add people, animals, or signs of life.

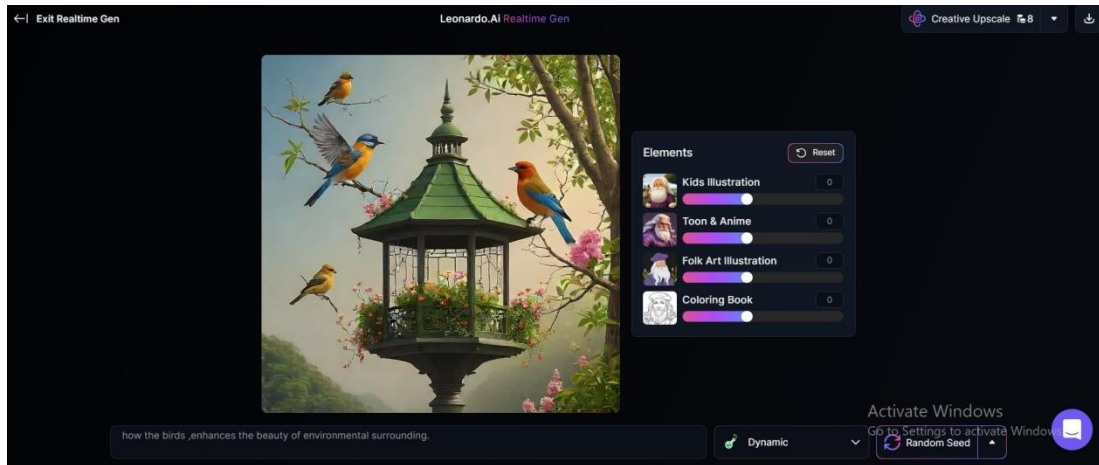
**Conclusion:**

The key to achieving a realistic scene is to provide detailed descriptions of the landscape, light, colors, and atmosphere, ensuring every sensory aspect of the scene is vividly described.



## Example 04:-

how the birds ,enhances the beauty of environmental surrounding.



Here is a Python code & process using the Leonardo AI:

```
import openai

# Define your OpenAI API key (replace with your actual API key)
openai.api_key = "your_openai_api_key"

# Define the text prompt for how birds enhance the environment
text_description = """
A beautiful, lush green forest with colorful birds flying around, some perched on trees.
The birds are pollinating flowers, eating insects, and singing melodiously, creating a peaceful environment.
Vibrant plants and trees are growing around, contributing to the lively surroundings.
"""

# Call the DALL-E model to generate an image based on the text description
response = openai.Image.create(
    prompt=text_description,
    n=1, # Generate one image
    size="1024x1024" # Image size
)

# Get the URL of the generated image
image_url = response['data'][0]['url']
print(f"Generated Image URL: {image_url}")
```

### 1: Identify Key Elements of the Scene:

Before inputting the text into Leonardo AI, you need to identify the core components that will form the image:

- **Environment:** Define the type of environment where the birds are present. (Is it a forest, park, lakeside, garden, or meadow?)
- **Bird Species:** Specify which birds are involved (e.g., hummingbirds, sparrows, parrots, or peacocks). This can affect the colors and appearance of the birds.

## **2: Craft a Detailed Text Prompt:**

A detailed and vivid description of the scene will guide Leonardo AI to generate an accurate image. When writing the prompt, consider including:

1. Scene Setup: Describe the type of environment (lush forest, peaceful garden).
2. Bird Appearance: Detail the birds' colors, size, and interactions with the surroundings.

## **3: Analyze the Visual Components:**

Once you have the prompt, break down the key visual elements to ensure the text is clear for AI interpretation:

- Color Palette: Are you emphasizing bright colors for the birds and flowers, or do you want a muted, natural look?
- Lighting and Shadows: Should the image have warm sunlight, soft shadows, or a misty glow?

## **4: Input the Text into Leonardo AI:**

Once the description is crafted, log into \*Leonardo AI\* (or any other similar text-to-image platform) and locate the \*Text-to-Image\* generation tool.

- Paste the Prompt: Enter the crafted prompt into the input field for text descriptions.

## **5: Generate the Image:**

After entering the text prompt and configuring any optional settings, proceed to generate the image.

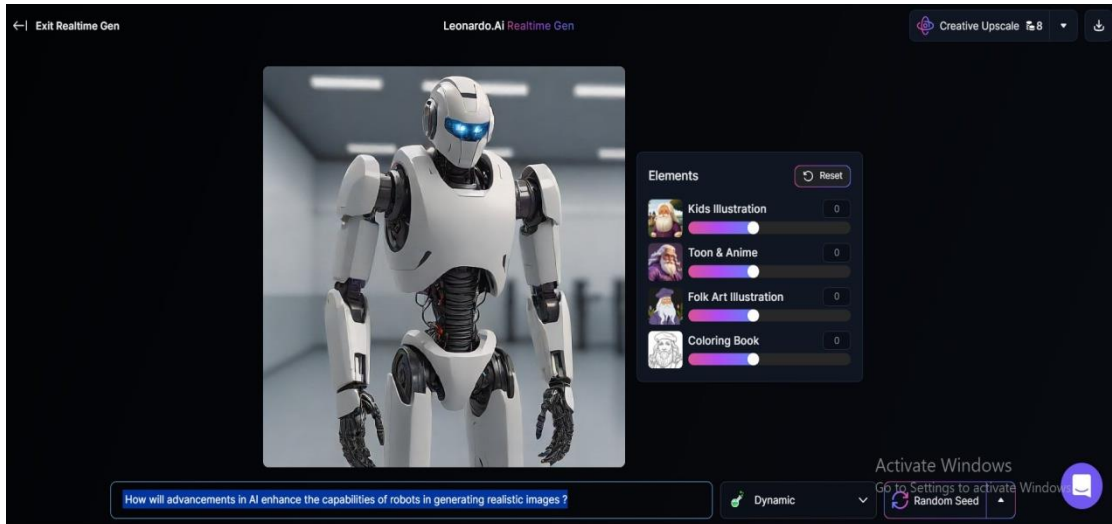
- Wait for Processing: Leonardo AI will take a few moments to interpret the text and generate an image based on the description.
- Review the Result: The AI will output an image that visualizes the text description provided. The quality of the image depends on the clarity and detail of your prompt.

## **Conclusion:**

**By following these steps, you can effectively use \*Leonardo AI\* to convert a text-based description of how birds enhance the beauty of their surroundings into a visually appealing image. The key to success lies in how well you craft and refine the text prompt to match your creative vision.**

## Example 05:

**How will advancements in AI enhance the capabilities of robots in generating realistic images ?**



**Here is a Python code & process using the Leonardo AI:**

```
import requests

def generate_image_from_text(description):
    # Hypothetical API endpoint for an advanced text-to-image AI model
    API_URL = "https://api.advanced-ai.com/generate" # Replace with actual endpoint
    API_KEY = "YOUR_API_KEY" # Replace with your actual API key

    # Set up the payload with the text description
    payload = {
        "prompt": description,
        "num_images": 1,
        "size": "1024x1024"
    }

    # Set up headers for authentication
    headers = {
        "Authorization": f"Bearer {API_KEY}",
        "Content-Type": "application/json"
    }

    # Send request to generate the image
    response = requests.post(API_URL, headers=headers, json=payload)

    # Check the response
    if response.status_code == 200:
        image_data = response.json()
        return image_data['images'][0] # Return the URL of the generated image
    else:
        print(f"Error: {response.status_code}, {response.text}")
        return None

# Example text description
description = "A futuristic cityscape with flying cars and vibrant neon lights."

# Generate the image
image_url = generate_image_from_text(description)
if image_url:
    print(f"Generated Image URL: {image_url}")
```

### Step-by-Step Process Analysis

- 1. Review Current Technology:**
  - Understand existing text-to-image models (e.g., GANs, diffusion models) and explore Leonardo AI's capabilities.
- 2. Integrate Advanced Algorithms:**

- Implement state-of-the-art deep learning models and enable continuous learning from user feedback.
- 3. **Enhance NLP Capabilities:**
  - Improve natural language understanding for better interpretation of complex prompts and contextual awareness.
- 4. **Increase Image Quality:**
  - Focus on generating higher resolution images with fine-tuning techniques for specific domains.
- 5. **Enable Real-Time Generation:**
  - Optimize processing for fast, real-time image generation and ensure scalability for multiple requests.
- 6. **Introduce User Customization:**
  - Allow users to customize image features and create an interactive feedback loop for ongoing refinement.
- 7. **Address Ethical Concerns:**
  - Implement strategies to mitigate bias and ensure copyright and originality in generated images.
- 8. **Develop APIs for Integration:**
  - Create user-friendly APIs to integrate Leonardo AI into various applications, including robotic interfaces.
- 9. **Conduct User Testing:**
  - Gather user feedback to refine models and enhance the overall experience through iterative development.
- 10. **Explore Future Directions:**
  - Stay updated with AI research for new models and investigate cross-modal capabilities to enrich image generation.

## **Conclusion:**

**By following these steps, you can analyze how advancements in AI, particularly through a platform like Leonardo AI, can significantly enhance the capabilities of robots in generating realistic images from text. This process not only improves the technical aspects of image generation but also focuses on user experience, ethical considerations, and future potential.**

# Usefull of using Leonardo AI Tool:

Leonardo AI offers several useful features and benefits, making it a valuable tool for various applications. Here are some key advantages:

## 1. Text-to-Image Generation

- **Creative Visuals:** Easily convert text prompts into high-quality images, useful for artists, designers, and marketers.

## 2. Customization

- **User Personalization:** Allows users to tailor images based on specific preferences or styles, enhancing creative expression.

## 3. Rapid Prototyping

- **Speed:** Quickly generate visual concepts, making it easier to iterate on ideas without extensive manual design work.

## 4. Enhanced Storytelling

- **Visual Narratives:** Supports authors and content creators by visualizing scenes or concepts from their narratives, enhancing storytelling.

## 5. Accessibility

- **User-Friendly Interface:** Designed for users without extensive technical skills, democratizing access to powerful AI tools.

## 6. Diverse Applications

- **Wide Range of Uses:** Applicable in various fields such as advertising, gaming, education, and product design.

## 7. Collaboration and Sharing

- **Ease of Sharing:** Facilitates collaboration by allowing teams to generate and share visual content quickly.

## 8. AI-Powered Insights

- **Trend Analysis:** Utilize AI to analyze and predict design trends, helping businesses stay relevant in competitive markets.

## 9. Integration with Other Tools

- **API Access:** Allows integration with other applications and workflows, enhancing productivity across platforms.

## 10. Continuous Learning and Improvement

- **Adaptive Models:** Benefits from ongoing improvements and updates, ensuring the technology evolves with user needs.

These features make Leonardo AI a versatile and powerful tool for anyone looking to leverage AI for visual content creation and beyond.

\*\*\*\*\*THE END\*\*\*\*\*