



The teaching crew — Ask the expert

By: Technical Skills Development Team

Learn Oracle from Oracle experts

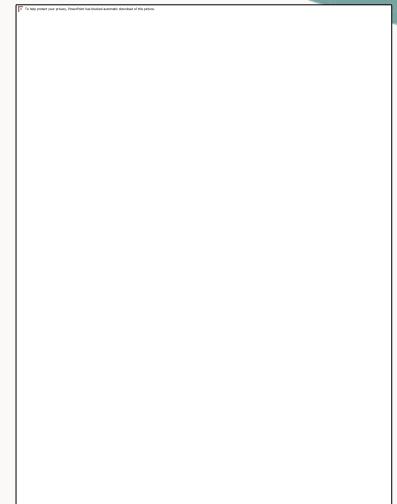
This is a unique program designed to boost employees' learning and skills by connecting with Oracle's in-house experts. Engage in tailored, interactive sessions where our specialists share their deep knowledge and expertise.

Name

Santhi Kondabolu

Oracle GSC

Oct 16, 2025



Speakers-virtual profile



Santhi Kondabolu

Functional Lead, Oracle



Experience

21 years



Expertise

Oracle Certified Consultant with 21+ years of expertise



Location

Hyderabad,
India

Building Generative AI Agents Using Prompt Engineering Day-1

Agenda title

- 1 Oracle Predictive Vs Generative AI
- 2 RAG
- 3 AI Agents
- 4 Prompt Engineering
- 5 Oracle AI Studio
- 6 Demo

Oracle AI for Fusion Applications?

Oracle AI for Fusion Applications refers to the AI functionality embedded within features in Oracle Fusion Cloud Applications. These features help you get business insights faster, automate routine tasks, access personalized recommendations, and improve business performance.



Key Focus Areas of Oracle AI for Fusion Applications

Focus Area	Description	Purpose
Embedded in Fusion Apps or AI services for self-build	AI features are embedded within Fusion Apps or available as AI services for custom builds.	To make AI work for the enterprise.
Data always private in Oracle Cloud Infrastructure	All AI operations happen within Oracle Cloud to ensure customer data privacy.	To protect customer data.
Intelligence to support human decisions	AI supports but does not replace human judgment in business processes.	So the human is always in the driver's seat.



Core Principles Illustrated



To make AI work for the enterprise

Embedded in Fusion Apps or AI services for self-build.



To protect customer data

Data always private in Oracle Cloud Infrastructure.

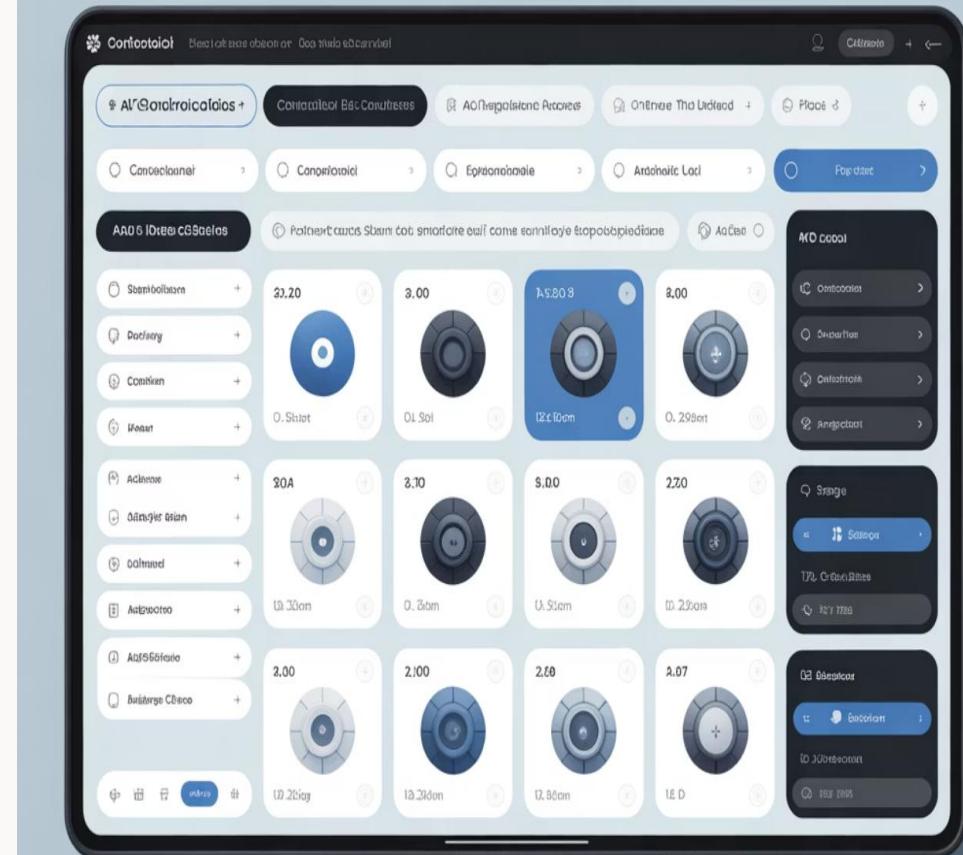


To support human decisions

Intelligence assists humans to stay in control.

Ease of Enablement

The features are prebuilt to include AI, and you don't need to have data science or machine learning expertise to use them. You can easily enable and set up these features, without any assistance from an external system integrator.



What are the technologies used in Oracle AI for Fusion Applications?

Oracle AI, in Fusion Applications, encompasses these types of AI technologies:

Predictive AI

Generative AI

- Language-based
- Context-aware
- Action-taking



What's Predictive AI?

Predictive AI identifies patterns in data and applies these insights to make predictions on new data. It typically involves machine learning models tailored to specific use cases, such as data classification, forecasting, and anomaly detection.

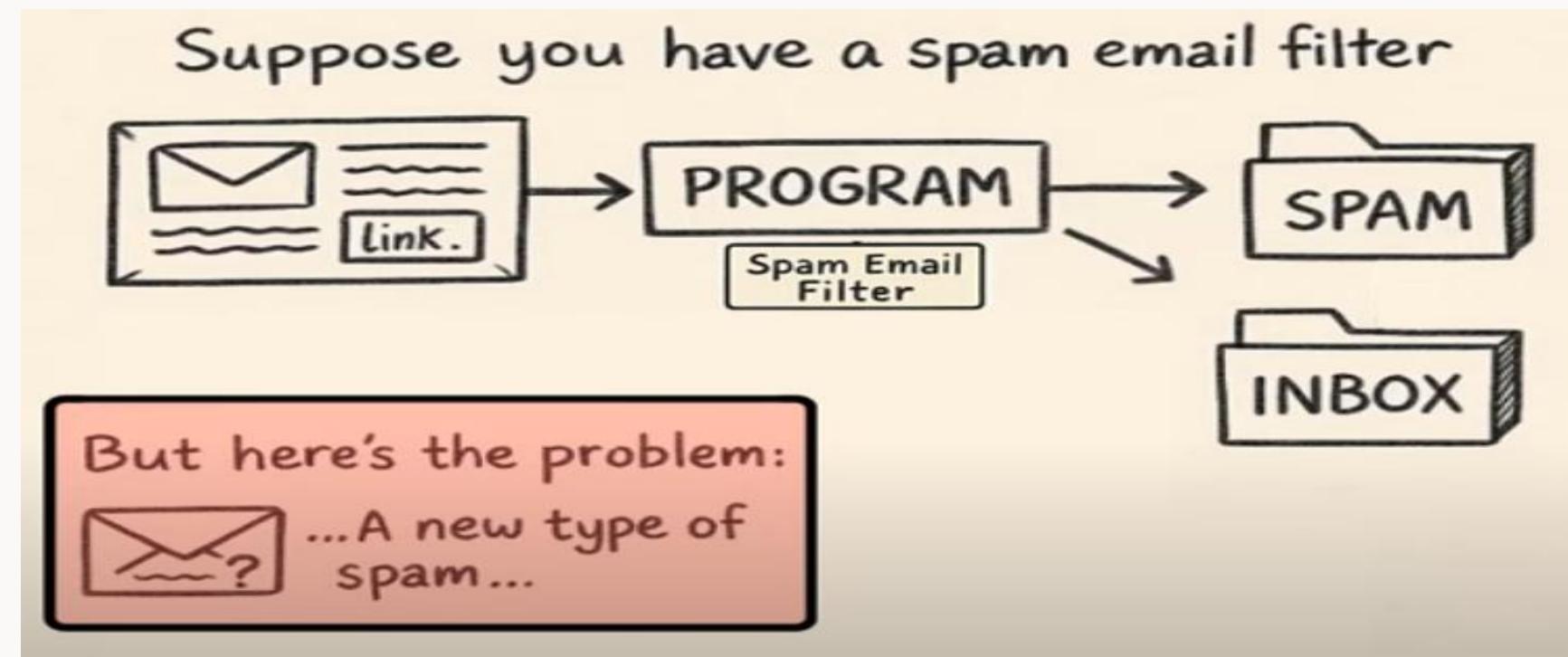
Predictive AI features use machine learning algorithms trained on each user's data, continuously learning and improving over time to deliver greater value and immediate benefits to users.

Examples include: summarizing a document, answering questions about your data, predictive cash forecasting, sales forecasting, project analysis, and so on.



Limitations of Traditional AI

- Traditional AI systems lack the ability to adapt and respond to new unknown scenarios (scenarios unseen during training) **without human intervention.**
- Example: Consider a ‘Email Spam Classifier’. If a new type of spam email is given to it, it will classify it as a valid email. To avoid this, we need to retrain it on the newer type of spam emails.
- Hence, traditional AI systems cannot adapt to new situations or learn from interactions without human intervention.



RAG (Retrieval Augmented Generation):

- Most of the LLMs are not trained on latest data.
- Also, LLMs are not trained on private data. E.g: Organizations internal data, etc.
- To solve these problems, we can finetune the LLM, but it is very expensive (time & money).
- Hence, we use RAG here.



What's Generative AI?

Generative AI features use LLMs to create content, including text, images, videos, and even code. Some common applications for text are assisted authoring, summarizing documents, and providing suggestions.

For example, you can generate the reply for an email and then review or edit the same before sending the reply.



Zoom AI Companion in Meetings

Boost Your Productivity and Collaboration

💡 Overview

Zoom AI Companion is designed to enhance productivity, facilitate collaboration, and capture key insights from every meeting. While not all AI features are available during the initial rollout, new capabilities will continue to be added.



Key Features



Real-time & Post-Meeting Summaries

Stay focused during discussions and revisit concise recaps afterward.



Automated Action Item Detection

AI identifies next steps and tasks from your meeting conversations.



Contextual Meeting Highlights

Catch up quickly if you join late or step away.



AI-Powered Note Taking

Concentrate on the discussion while AI captures your key points automatically.

Enabling AI Companion Features

To enable and customize Zoom AI Companion:

01

 Open the Zoom Web Portal in your browser.

03

 At the top, select AI Companion.

 Benefit: Save time, reduce manual note-taking, and improve collaboration — letting AI handle the details while you focus on what truly matters.

02

 On the left panel, click Settings.

04

 For features that can be customized at the user level, select your preferred settings here.



Tool Spotlight: Oracle Code Assist (OCA)

Empowering Oracle Engineers with AI-Driven Coding

Overview

Oracle Code Assist (OCA) is Oracle's AI-powered coding assistant, designed to enhance developer productivity and code quality. It seamlessly integrates with your favorite development environments to help write, optimize, and understand code faster.

Where You Can Use It

-  IntelliJ IDEA
-  Visual Studio Code (VS Code)
-  Command Line Interface (CLI)

Key Capabilities



Code Suggestions & Autocompletion

Get intelligent recommendations while coding.



Code Explanation & Optimization

Understand existing code and identify improvements.



Error Detection

Spot and fix issues in real time.



Natural Language to Code

Describe logic in plain English and let AI generate the implementation.

Learn More

For detailed documentation, demo videos, and setup instructions, visit:

 [AI for Engineering – Oracle SharePoint](#)

Why Agents?



Larry Ellison (Executive Chairman & CTO) - “All of our applications are becoming AI agents.” “You won’t be able to separate how much you made on the AI agents and how much on the rest of the application. The applications themselves will migrate to be basically a bunch of connected AI agents.”



Steve Miranda (EVP of Applications, Oracle) When announcing AI Agent Studio, Miranda said: “AI agents are the next phase of evolution in enterprise applications ... business leaders need the flexibility to create functionality that addresses evolving business needs.”



Just look at Bill Gates. In his blog, he wrote: “In the computing industry, we talk about platforms—the technologies that apps and services are built on. Android, iOS, and Windows are all platforms. Agents will be the next platform.”

Flavors of AI Agents

- **Embodied Agents:**

- Interact with physical or simulated 3D environments
- Used in robotics, autonomous navigation, and gaming
- Rely on sensors, cameras, real-time decision-making
- Trained using reinforcement learning in dynamic environments

- **Software Agents:**

- Operate in digital environments (e.g., emails, workflows, office tools)
- Automate repetitive tasks, act as intelligent assistants
- Trained using large datasets and LLMs for humanlike interaction



Types of Generative AI Agents

Generative AI also includes various types of agents, which work together to help you create and interact with content in a dynamic and responsive way.

Type of Agent	Description	Example
Language-based	Interacts with you using human languages.	Chatbots, virtual assistants.
Context-aware	Always on, sensing and reacting to data from your environment, even as the data changes.	If a user in your organization asks about their medical data, the agent can pull the relevant details directly from your organization's database.
Action-taking	Takes action automatically, with you guiding and approving in key moments. These agents can autonomously complete tasks, use tools, and loop you in to review or approve	If you request a vacation booking, the agent might suggest possible vacation dates based on your calendar, find the hotels with vacancy, and create a booking with your approval.



Agent Use Case - Personal Assistants

Examples: Siri, Alexa, Google Assistant.



Understand queries

Process and comprehend natural language inputs from users



Manage tasks across time

Handle calendar events, set reminders, and track commitments



Retrieve information

Access and utilize various skills and tools to find relevant data

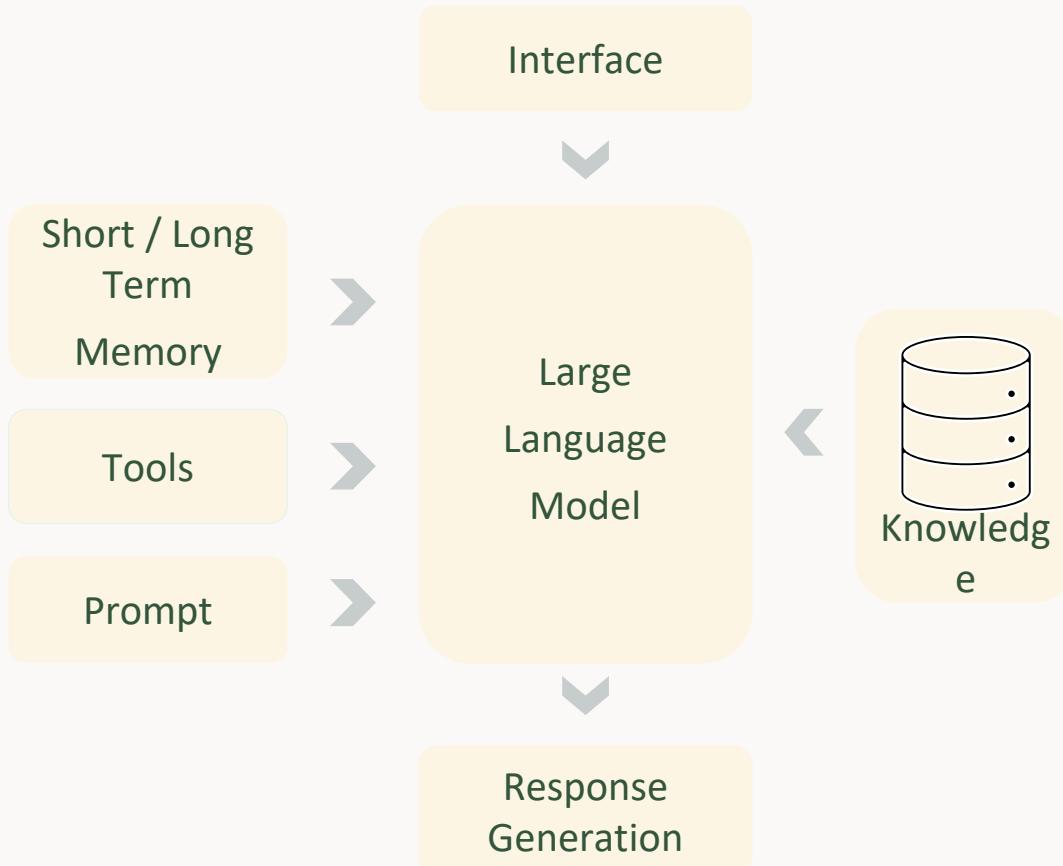


Act on user's behalf

Make calls, send messages, and perform actions as requested

Learning from feedback and adapting behavior over time.

What is an AI Agent



An AI AGENT is an LLM-based application that can:

- Perform complex tasks on its own
- Mimic human chain-of-thought processing
- Be an effective tool for automating processes
- Utilize your knowledge

Why AI Agents

ReAct Framework

- Reasoning and Action

Enhanced efficiency

- Reduced the time and effort required to complete them

Error Reduction

- Reduced manual mistakes

Tangible business outcome

- can lead to significant cost savings, increased productivity, and enhanced customer experiences

Data Driven Decisions

- Analyze information and recommend actions

Defining the Problem for Agentic AI

Frame the task as a goal

Define a goal that requires multiple steps, decisions, or data sources.

Identify key components

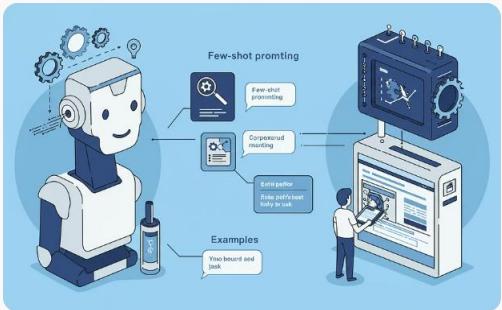
Determine inputs, desired outputs, constraints, and edge cases.

Assess suitability

Ask: Can this problem benefit from autonomy, memory, or tool use?

Example: "Close all inactive projects after validating billing, revenue, and cost completion."

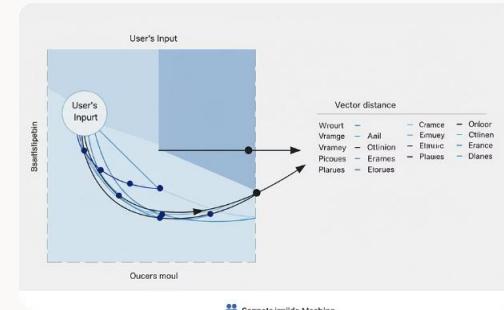
How Are Tools/Skills Selected Internally?



Generative Skill Selection

Few-shot prompting to suggest tool usage.

Combine both: generate, then re-rank semantically.



Semantic Skill Selection

Embedding similarity between user input and skill metadata.

Tool Topologies

Single Skill Execution
One tool used at a time (sequential planning).

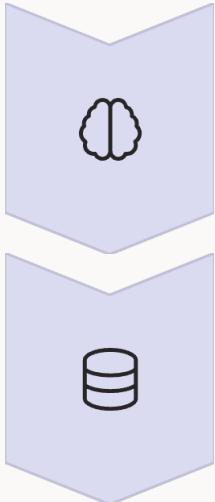


Parallel Skill Execution
Multiple tools invoked together; useful for retrieval, summarization.

Chain-of-Tools
Output of one tool passed as input to another.



Agent Memory - Short-Term vs Long-Term



Short-Term Memory

Maintains current task context (e.g., conversation history, goal).

Long-Term Memory

Persistent knowledge base (e.g., embeddings of documents, CRM data).

Architecture:

- Retrieval Augmented Generation (RAG)
- Embedding search + summarization

Single-Agent Architecture

One agent performs planning and execution.

Simplifies coordination, good for linear workflows.

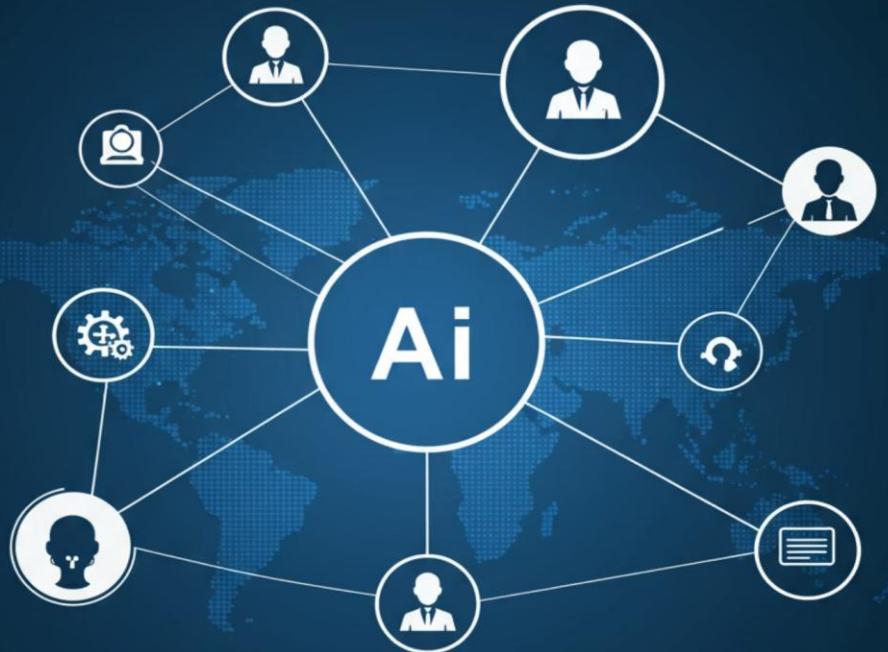
Risks:

- Overload of capabilities
- Harder to scale or maintain



Advantages of Single Agent Systems

- Easier to orchestrate and manage.
- Suitable for simple workflows.
- Lower cost, fewer infrastructure needs.



Multi-Agent Architecture

- Multiple agents, each with a specialized role (planner, executor, retriever, validator).
- Can communicate asynchronously (event-driven).
- Better modularity, reuse, and robustness.

Traditional Software vs Generative AI Agent Development

1. Traditional Software Development

- Deterministic Approach: Predictable and repeatable outcomes.
- Linear Workflow: Phases like requirement gathering, design, development, testing, and deployment follow a defined path.
- Clear Documentation: Facilitates team coordination and scalability.
- Ideal for: Large teams and complex, rule-based projects.

2. Generative AI Agent Development

- Probabilistic Nature: Outputs vary even with the same input; non-deterministic.
- Use Case Identification: Challenging to find tasks suitable for AI (especially those needing predictable behavior).
- Model Selection: Complex due to frequent model updates and high sophistication.
- Cost Considerations:
 - API usage charges or
 - Hardware expenses (e.g., GPUs for local deployment)
- Complex Workflows: Require careful evaluation due to inherent AI variability.
- Mitigation Strategies:
 - Implementing guardrails
 - Human-in-the-loop mechanisms



Performance Evaluation

Evaluate via:

- Task success rate
- Latency (response time)
- Number of tool calls per task
- Consistency across repeated prompts
- Human feedback scores (ratings, flags)
- Handling Unexpected Inputs



Challenges in Agentic AI

Inconsistency in behavior

Agents may respond differently to similar inputs across sessions.

Hallucinations

False responses that appear plausible but are factually incorrect.

Feedback integration

Challenges in effectively learning from user corrections.

Handling ambiguity

Struggling with unexpected or unclear inputs.



Best Practices for Designing Agents

1

Scope tasks clearly

Define boundaries and goals with precision, Do not overload with too many details

2

Use modular skills/tools

Build reusable components that can be combined flexibly, Define Output formats

3

Design memory for both context and learning

Implement effective short-term and long-term memory systems

4

Implement feedback loops

Include mechanisms for rating and correction

5

Secure, testable, observable design

Ensure the system can be monitored and validated



Best Practices for Designing Multi-Agent Systems

1. Roles of Agents:

- Supervisor Agent -Understands user intent and context
 - Delegates tasks to the right worker agents
 - Coordinates multi-step conversations
 - Oversees the overall workflow and agent orchestration
- Worker Agent Handles specific tasks or focuses on defined functional domains

2. Agent Partitioning Strategies:

- Domain-Based Partitioning: Assign agents to business areas like Recruitment, Learning, Payroll, etc.
- Function-Based Partitioning: Group agents by task type, e.g., payslip queries, skill updates, personal info changes.



3. Why Partitioning Matters:

- Focused Expertise: Each agent specializes in a product or domain (e.g., HCM).
- Higher Accuracy: More precise and relevant responses.
- Efficient Tool Management: Tools can be grouped and reused logically.
- Clear Escalation Paths: Enables smooth handoffs to other agents or humans when needed.

Tip for Learners

"Start small — build your own personal agent that automates a task you do daily (like Data Retrieval, Data summarization, or Data Updating). That's how real innovation begins."



Overview of Frameworks

LangGraph

Graph-based, stateful flows

Autogen

Multi-agent chat and tool coordination

CrewAI

Team-based collaborative agents

Oracle AI Agent Studio

Enterprise-grade Fusion integration and security



When to Use Which Framework

Use Case	Framework
Research / Simulation	Autogen / CrewAI
Workflow automation	LangGraph
Oracle ERP/PPM/HCM extensions	Oracle AI Agent Studio
Prototyping / Teaching	CrewAI / LangGraph



What Is Prompt?

Definition

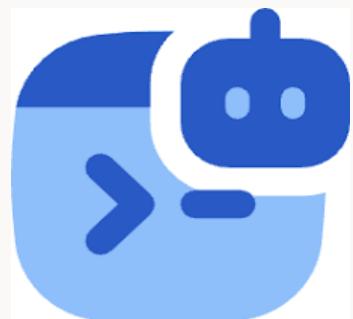
A prompt is the input text given to an AI model that guides it toward generating a specific output. Think of it as a conversation starter that sets the direction and boundaries for the AI's response. Prompts can range from simple questions to complex instructions with multiple components.

Components

Effective prompts typically contain three key elements: context (background information that frames the task), instruction (specific directions about what the AI should do), and examples (demonstrations of desired outputs). The balance of these elements significantly impacts the quality of AI-generated results.

Purpose

The primary purpose of a prompt is to bridge the communication gap between human intent and machine understanding. A well-crafted prompt serves as a translator, converting our goals into language that the AI can interpret and act upon accurately.



Types of Prompts

Type	Description	Example
Zero-shot	Instruction only	"Summarize this report."
Few-shot	With examples	Show Q&A pairs before asking
Chain-of-thought	Step-by-step reasoning	"Explain before answer."

Prompt Engineering Techniques



Role-playing

Assigning a specific role to the AI can dramatically improve outputs for specialized tasks. For example: "Act as an expert physicist explaining relativity" helps the model adopt the appropriate tone, depth, and vocabulary. This technique works by leveraging the model's training on texts written by experts in various fields.



Temperature Adjustment

Temperature controls the randomness in AI responses. Lower values (0.0-0.5) produce more deterministic, focused outputs ideal for factual tasks. Higher values (0.7-1.0) introduce more creativity and variability, better for creative writing or brainstorming where diverse options are desired.



Iterative Refinement

The most successful prompt engineers rarely get perfect results on the first try. This technique involves gradually refining prompts based on the AI's responses, adding constraints or clarifications until the output meets your needs. It's an experimental process that improves with practice.

Best Practices for Creating Prompts

Best Practice	Description
Start Simple and Iterate	Begin with a basic prompt, review the AI's output, then refine it gradually for improved accuracy and clarity.
Experiment with Prompt Structure	Try different word orders, spacing, and placement — small structural changes can greatly affect results. <ul style="list-style-type: none">• Change word order to test variation• Adjust line spacing• Move information placement to see its effect
Use Detailed Commands	Use action words such as Write, Summarize, Translate, Order, or Classify to give clear task instructions.
Be Specific	Provide clear, descriptive instructions to guide the AI. Specificity ensures more consistent and accurate outputs.
Be Mindful of Length	Keep prompts detailed but concise — long, unfocused prompts can confuse the model or waste tokens.
Specify the Output Format	Define exactly how the output should look (e.g., HTML, table, JSON) for Oracle Fusion Cloud Applications or reports.
Avoid PII	Never include personally identifiable information in prompts — maintain data security and compliance.
Test Your Prompts	Try both positive and edge-case (happy/sad path) scenarios to validate that the prompt stays accurate and reliable.



Best Practices

Be clear and specific

Provide context & constraints

Use role prompting ("You are an Oracle PPM expert...")

Define output format (tables, JSON, etc.)

Include few-shot examples



Prompt Debugging



Modify one variable at a time



Compare outputs for drift



Use self-reflective prompts



Track prompt versions & performance

Hands-on with Oracle Generative AI Chat

Learn how to apply Prompt Engineering using Oracle's internal GenAI platform

- ◊ Tool: <https://chat.oracle.com>
- ◊ Access: Oracle SSO + MFA
- ◊ Goal: Practice safe, effective, and real-world prompt techniques





Accessing Oracle Generative Chat

01

Open your browser → go to
chat.oracle.com

02

Log in using two-factor authentication
(SSO required)

03

You'll see the LLM selector at the top-left corner



Tip: Always start with the default model (GPT-4.1) for balanced performance.



Choosing the Right LLM Model

Model	Best For	Notes
GPT-4.1	Complex technical tasks, coding, professional writing	Default, most accurate
OpenAI o3	Analytical and reasoning-heavy tasks	Strategic planning
GPT-4o-search-preview	Real-time web data (non-Oracle only)	Internet enabled
xAI Grok 3	Conversational & creative content	Brainstorming, storytelling

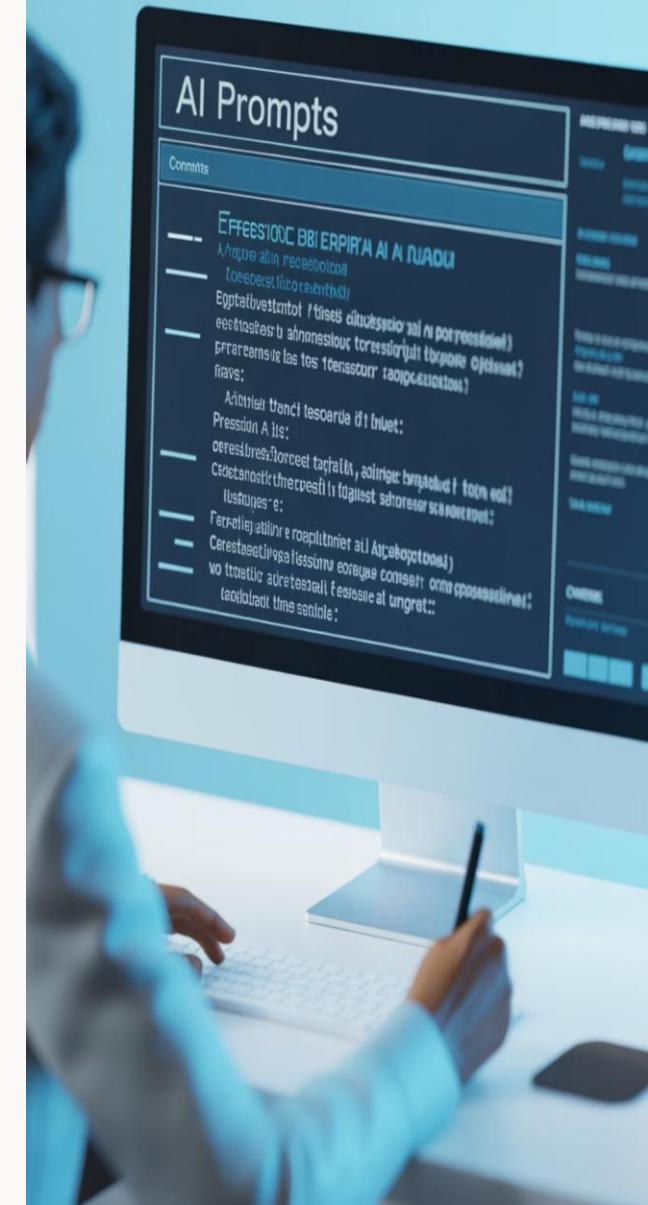
 Select model → click on dropdown → type your prompt.

Writing Effective Prompts

Prompt = Your instruction to the model. Better prompts = Better results.

Follow these steps:

- 1 Define context → "You are an Oracle consultant..."
 - 2 Specify goal → "Generate a summary of this document..."
 - 3 Add format → "Return output in bullet points/table."
 - 4 Refine → "Use professional tone, limit to 150 words."
-  Reference: Oracle Prompt Library (VPN required)

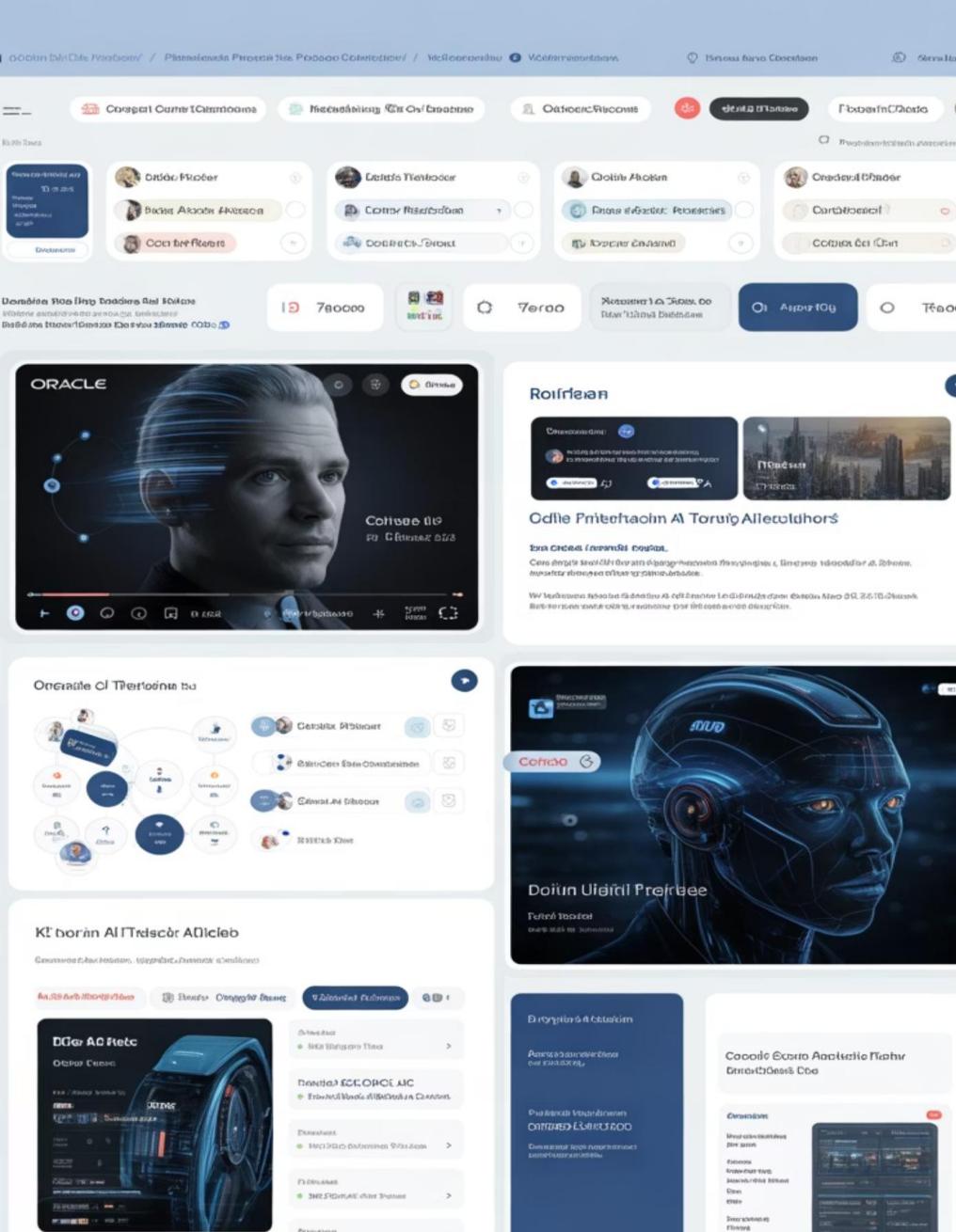


Hands-On Activity

Try these in chat.oracle.com 

Exercise	Example Prompt
Summarization	"Summarize this uploaded project charter in 5 bullet points."
Question Answering	"Explain the key features of Oracle Project Financial Management in simple terms."
Code Generation	"Generate a Python function to calculate project budget variance."
Creative Task	"Draft a professional email to request project milestone approval."

 Observe tone, structure, and accuracy. Then refine your prompt.



Explore More Videos

This link also includes other AI in Action tutorials, such as:

- Summarizing a document
- Comparing documents
- Creating a project plan from documents
- Creating user stories from documents
- Creating risks from documents
- Building your personal prompt library
- ...and many more Oracle GenAI use cases

<https://oracle.sharepoint.com/sites/ai-for-employees/SitePages/Gen Chat Videos.aspx>

Best Practices & Guidelines

Combine AI output with your expertise

Verify key facts from trusted Oracle sources

Avoid confidential or sensitive data

Use AI for brainstorming, not decision-making

Review before publishing or sharing outputs

 Remember: Prompt Engineering = Iteration + Context + Clarity.

Working with Oracle Prompt Library

Introduction

The Oracle Prompt Library is a curated repository where employees can explore, reuse, and refine high-quality prompt templates for generative AI tasks.

How to Use It

- Browse prompt categories (summarization, coding, analysis, emails, etc.)
- Copy a prompt and adapt it to your use case
- Combine with your context for better output
- Always test and iterate prompts before deploying

Watch Demo Video

Below is a sample video demonstrating how to work with Oracle's Prompt Library:

 [AI in Action – Working with Oracle's Prompt Library](#) (Click the link to view the video demonstration.)



How to Add to Oracle Prompt Library

Introduction

The Oracle Prompt Library allows users to contribute their tested prompts so others can reuse and refine them.

Steps to Add a Prompt (Bulleted Guide)

- Log in to the Oracle Prompt Library (VPN access may be required)
- Navigate to "My Prompts → Add Prompt"
- Enter Prompt Title, Description / Use Case, Prompt Text, Example Input/Output
- Tag with relevant categories (e.g. summarization, coding, analysis)
- Submit the prompt for review (Oracle team will validate before publishing)

Watch Demo Video

 [AI in Action – Adding to Oracle's Prompt Library](#) (Click the link to view the step-by-step video.)

Grow Your AI Skills with Oracle AI Training Programs

Access Point

[AI Training – Oracle SharePoint](#)

Explore AI in Training resources and the Artificial Intelligence Skill Plan (Beginner, Intermediate, Advanced).

Recommended Learning Path

How to Boost Your Productivity with AI Tools – LinkedIn Learning (1h)

- Learn to integrate AI tools in daily workflows
- Explore writing, research, and task automation with AI
- Gain practical prompting techniques and overcome AI resistance

How to Use Generative AI at Work – O'Reilly (1h 23m)

- Practical ways to use AI tools like ChatGPT (chat.oracle.com)
- Real-world use cases: project management, communication, finance, and coding

AI for You – Oracle University (3h)

- Fundamentals of AI, ML, Deep Learning
- Data Science, and AI Agents
- Oracle's AI capabilities across Fusion Applications
- Buy vs. build AI decisions
- Understand risks, biases, and ethics in AI

Prompt Engineering (L&D Blog)

- Learn to design effective prompts for accurate AI outputs
- Useful for developers, PMs, data scientists
- content creators



Quiz



Q & A

Poll - Time



ORACLE