

**Scan to ask questions**





CURSOR

RIYADH

# Build your MVP with AI!

July 30, 2025

Cursor Riyadh

## Our Partner



مركز ريادة الأعمال الرقمية  
CENTER OF DIGITAL ENTREPRENEURSHIP

Hosting Partner

# Agenda

6:00pm  **Introduction** 30 min

Choose partners, choose problems to solve

6:30pm  **BUILD TIME!** 2.5 hours

Build your MVP with AI tools - we're here to help!

9:00pm  **Presentations** 30 min

Show off your creations to the group

9:30pm  **Wrap Up**

# Meetup Introduction



Setting the stage for our Cursor journey

What is vibe coding?



NIK ✅ @ns123abc · Jul 29

vibecoding =  
prompt engineering  
context engineering  
your own knowledge  
skill  
taste



Oregon State  
University



deep.са

Malaa

## Mazen Alotaibi

AI Staff Engineer at deep.sa

Senior ML Engineer at Malaa Technologies

MSc AI and BSc CS from Oregon State University

2023 PyTorch Ambassador Award & Cursor Ambassador

Accelerating technology and AI development in

ma7.dev | @ma7dev



CURSOR

RIYADH



## Faris Hijazi

Data Science Lead Specialist at Thiqah

BSc CEng from King Fahd University of Petroleum and Minerals

Faris Hijazi | @theefaris

+966 505501494



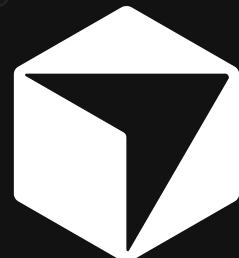
# Cursor

## **Anysphere** (parent company)

- \$9B valuation (May 2025)
- \$900M funding raised
- Fastest-growing AI startup

## **Cursor** (the product)

- AI-native code editor
- Built on VS Code foundation
- ~1B lines of code generated daily



# Why This Meetup?



## Knowledge Sharing

Learning from global best practices



## Developer Community

Building a strong local AI development ecosystem



## Innovation Hub

Positioning Riyadh as an AI development center



## Vision 2030

Aligning with Saudi Arabia's tech transformation

# What to expect

## In-person

- Talks by local experts
- Hands-on workshops
- Networking opportunities

## On Discord

- Sharing knowledge, tips, and tricks
- Virtual hackathons (with prizes)
- Community discussions

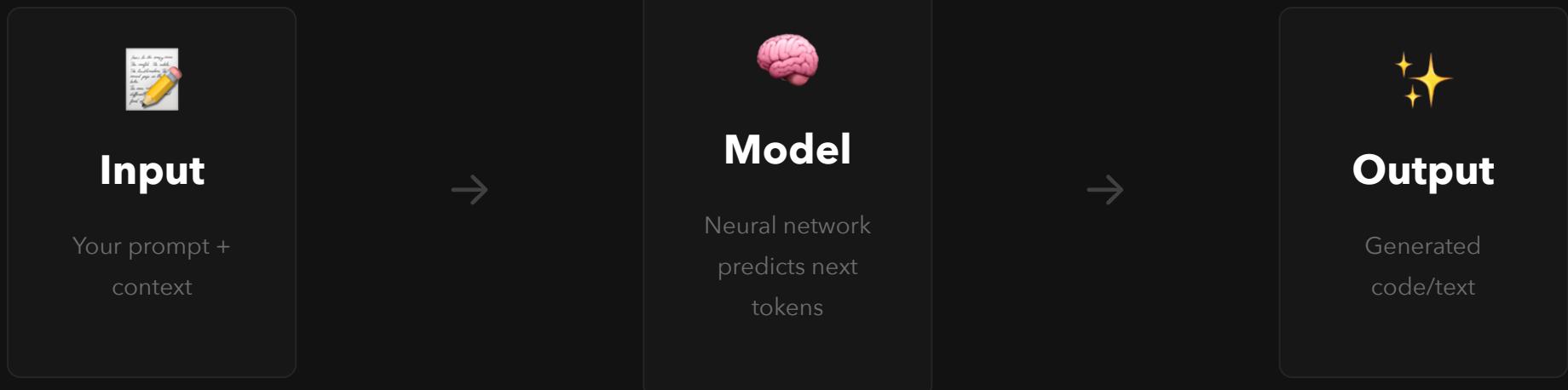


# Everything About Cursor



Deep dive into Cursor's capabilities and practical usage

# How LLMs Work (The Basics)



# LLMs in Code Editors Made Simple

## You Say:

"Make this button green"

## AI Sees:

Your code + files + request

## AI Thinks:

"Based on patterns, this needs CSS changes"

## AI Delivers:

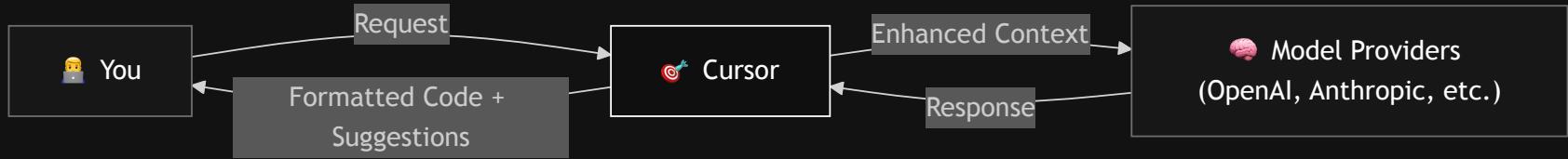
Updated code with green button

## The Secret

Better Context = Better Results

- Include relevant files
- Describe your goal clearly
- Show error messages
- Provide examples when helpful

# Cursor: The Smart Middle-Man



# Core Cursor Features

## ★ Cursor Tab

Next-gen autocomplete

- Multi-line suggestions
- Cross-file awareness
- Fusion Tab Model

## ▀ CMD+K

Inline code magic

- Quick edits
- Natural language commands
- Instant refactoring

## 💬 Chat

AI coding partner

- Four powerful modes
- Contextual assistance
- Background agents



## Agent Mode

- Autonomously learns codebase
- Makes codebase-wide changes
- Background processing
- Perfect for large refactors



## Manual Mode

- Focused edits with your context
- Full control over AI visibility
- Perfect for sensitive code
- Precise, controlled assistance

## ? Ask Mode

- Get code explanations
- Plan new features
- Understand complex logic
- Educational and exploratory

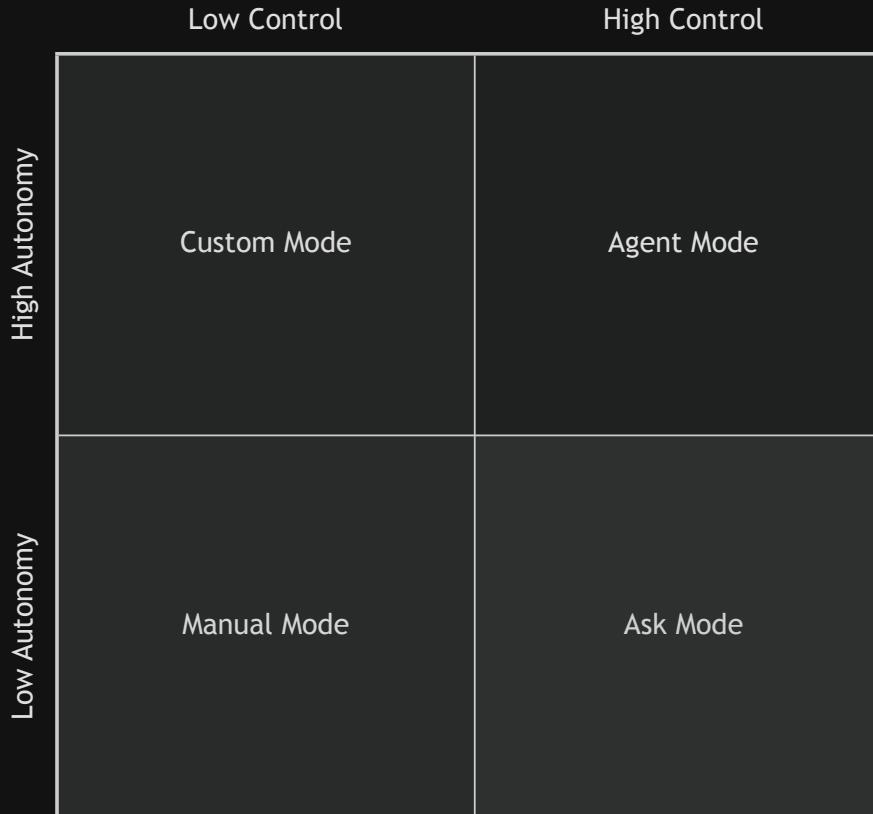


## Custom Mode

- Create workflow-specific modes
- Tailor AI behavior to team needs
- Unlimited customization
- Enterprise-ready configurations

# Chat Modes Deep Dive

Chat Modes by Autonomy vs Control



# Features

-  **Context Enhancements:** Smarter file and directory inclusion
-  **Docs:** Enable AI to use human-readable documentation as context
-  **Rules:** Set up user-specific or project-specific rules to guide AI behavior
-  **AI Commit Messages:** Intelligent git commit generation
-  **Checkpoints:** Restore previous states easily
-  **Multi-Root Workspaces:** Manage multiple repositories with cross-folder indexing
-  **Model Context Protocol (MCP):** Interact with external tools
-  **Background Agent:** Parallel task execution for larger tasks (early preview)
-  **Bugbot:** Automated bug detection and fixing
-  **PR Index:** Pull request analysis and management (Ultra plan)
-  **Enhanced Tab Model:** Multi-file edits, refactoring, and navigation
-  **Refreshed Inline Edit:** Improved CMD+K with new UI and options
-  **Max Mode:** Token-based pricing for complex tasks requiring extensive context
- and much more...

# Live Demo

Cursor in action

# Common Pitfalls to Avoid

1. **✗ Vague requests** without sufficient context
2. **✗ Ignoring error messages** and logs in your requests
3. **✗ Not using project rules** to guide AI behavior
4. **✗ Overwhelming AI** with irrelevant information
5. **✗ Expecting perfection** on first try - iterate!
6. **✗ Not exploring models** - each has its strengths

# Best Practices & Tips

1.  **Context is King:** Provide clear, relevant context
2.  **Provide error messages:** AI can't fix what it doesn't know about
3.  **Use Rules:** Set up project-specific guidelines
4.  **Leverage @ Symbols:** Be explicit about what you want
5.  **Iterate:** Start broad, then refine your requests
6.  **Experiment:** Try different models for different tasks

# Pricing



**Hobby**

Free



**Pro**

\$20/month



**Pro+**

\$60/month



**Ultra**

\$200/month



**Teams**

\$40/user/month

Cursor for Students: Free for 1 year with academic email

Enterprise plans available with custom pricing | Source: Cursor.com, July 2025

# Memory Bank & Advanced Usage



Using Cursor for everything - Demo by @ma7dev

# This Section Has Two Parts



## Memory Bank

Giving Cursor Agent the steering wheel  
(vibe-coding)



## Advanced Usage

Using Cursor for everything  
(beyond coding)

# What is Memory Bank?

## Concept

- Inspired by Cline Memory Bank
- Gives LLMs more context to plan and act
- Human-readable AI contract
- Updates and evolves over time

## Purpose

- Persistent project knowledge
- Context that survives sessions
- Intelligent task planning
- Continuous learning from progress

# Memory Bank: Three Parts



## Setup

Define project foundation



## Planning

Initialize & review progress



## Execution

Implement & update

# Part 1: Setup



## Foundation Building



### Define Project Description

- Project goals and objectives
- Technical requirements
- Architecture decisions
- Team conventions



### Iterate & Refine

- Use Cursor Agent to help
- Refine based on feedback
- Update as project evolves
- Keep it human-readable

# Part 2: Planning

## Strategic Thinking

### Model Choice

- **o3** (expensive but powerful)
- **Sonnet-4** (balanced option)
- Choose based on complexity

### Process

1. Initialize memory bank with Agent
2. Review memory bank content
3. Iterate & refine memory bank (especially progress)

# Part 3: Execution (Sonnet-4)

## 1. Start Fresh

Always start with a new chat

## 2. Implement and interact w/ AI

- Start with progress context
- Review changes actively
- Interact with AI

## 3. Complete & Update

- Review git changes
- Take code ownership
- Update memory bank before next task

# Live Demo

Memory Bank in Action

# Using Cursor for Everything



## Development

- Code generation & refactoring
- Bug fixing & debugging
- Documentation writing
- Test creation



## Beyond Code

- Technical writing
- Architecture planning
- Code reviews
- Learning new technologies



## Team Workflows

- Onboarding new developers
- Code style enforcement
- Knowledge sharing
- Project standardization



## Advanced Usage

- Custom model integration
- Workflow automation
- Multi-project management
- Performance optimization

# Next Steps & Resources



## Try Cursor

Download and start your free trial



## Join Community

Discord, forums, local groups



## Learn More

[docs.cursor.com](https://docs.cursor.com) for comprehensive guides



## Future Meetups

Monthly sessions with advanced topics

Thank You! 🙏

**Let's Connect & Build Together**

@ma7dev & @theefaris

Follow for more AI development insights



Networking Time! 🤝

**Let's build the future together**



Cursor Official



Cursor Meetup Riyadh