AI Agents

A brief primer

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

AI agents are...

“...software systems that use AI to pursue goals and complete tasks on behalf of users. They show reasoning, planning, and memory and have a level of autonomy to make decisions, learn, and adapt.”

(Google)

Shared under Build Club’s learning commons. Built for the community. Please share, not sell.

AI agents are...

"...autono *t, make decisions* *t human oversight"*

...chatbots that can push buttons

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

	AI Agent	AI Assistant	Bot
Purpose	Autonomously and proactively perform tasks	Assisting users with tasks	Automating simple tasks or conversations
Capabilities	Can perform complex, multi-step actions; learns and adapts; can make decisions independently	Responds to requests or prompts; provides information and completes simple tasks; can recommend actions but the user makes decisions	Follows pre-defined rules; limited learning; basic interactions
Interaction	Proactive; goal-oriented	Reactive; responds to user requests	Reactive; responds to triggers or commands

(Google)

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

Computers have a 1D view of the world

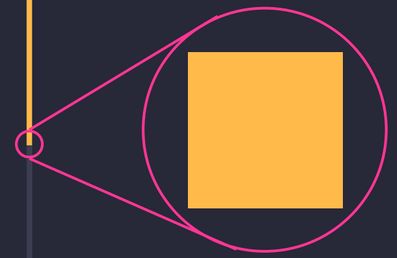
“Draw a square”

Shared under Build Club’s learning commons. Built for the community. Please share, not sell.

Computers have a 1D view of the world

“Draw a square”

What you see



One pixel at a time

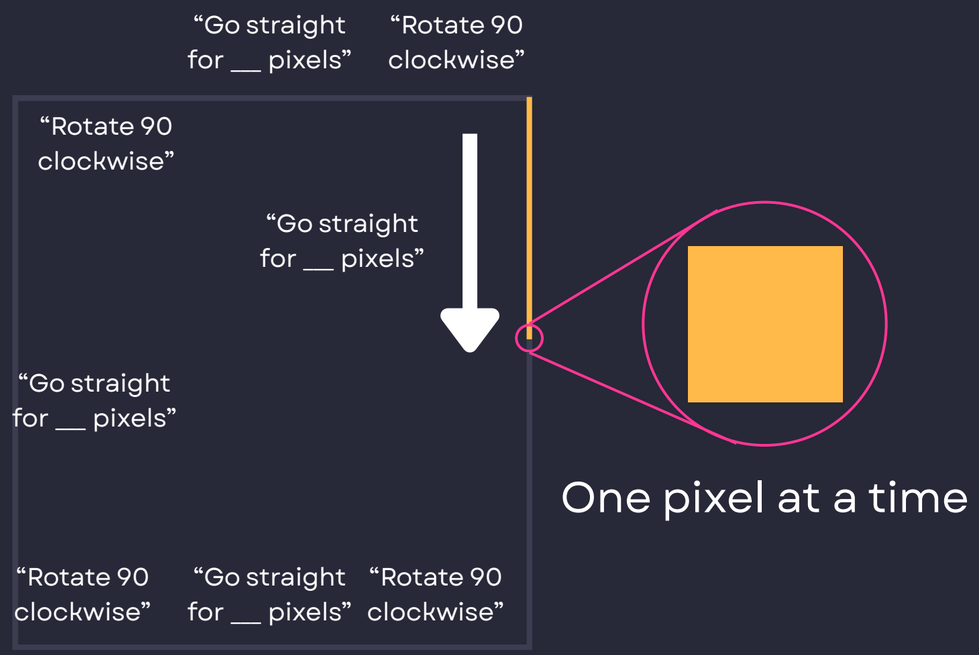
What they see

Shared under Build Club’s learning commons. Built for the community. Please share, not sell.

Computers have a 1D view of the world

"Draw a square"

What you see



What they see

Computers cannot see the whole screen

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

Humans are more flexible at solving open-ended problems

Imagine going grocery shopping if you could only see a small part of the world at any given time

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

Imagine going grocery shopping as a computer

Problem: Get groceries

↓

Choice: which store?

Tesco, *Aeon*, *99*, *7-11*, ...

↓

Choice: how do we get there?

Car, *walk*, *bicycle*, *airplane*, ...

↓

Choice: what if the door is closed?

What if it's a tunnel instead?

↓

...

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

Conditions + logic → smarter decisions

Problem: Get groceries

Choice: which st

Tesco, A

get there?

, bicycle, airplane, ...

Choice: what if the door is closed?
What if it's a tunnel instead?

...

We use **control logic** to decide what to run based on conditions

E.g.

```
def choose_store(store):
     if tesco.status == "open":
          store = tesco
      elif aeon.status == "open":
          store = aeon
       else:
          store = 99

     if store == tesco:
          while x in range (len(nkve)):
               x+=1
```

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

LLMs + programs = AI Agents

```
def next_action(state):
        prompt = {
                "You are a troubleshooting agent."
                f"Allowed actions: {', '.join(PERFORMANCE_ACTIONS)}."
                f"Current state -> disk_space={state['disk_space']}%, "
                f"high_cpu={state['high_cpu_processes']}. "
                "Respond with the format:\nAction: <one_action>\n"
        }


        reply = llm_chat(Prompt)
        Return reply.split(":",1)[1].strip()
```

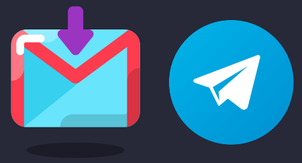
LLM decides on course of action based on input data

Programs act as tool calls to execute given actions based on decision

```
def handle_performance_ticket(state):
        action=next_action_llm(state)
        if action == "clean_temp_files":
                clean_temp_files()
        elif action == "kill_high_cpu_processes":
                kill_high_cpu_processes()
        elif action =="escalate_deep_malware_scan":
                escalate("deep malware scan")
        elif action == "close_ticket":
                close_ticket()
```

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

AI Agents are comprised of five parts



Perception / input

Environmental triggers, e.g. messages, API calls, sensor data

Decision-making logic

- Determine what actions need to be performed at what stage.
- Manage through an AI model, typically an LLM.

Function calls (tools)

- LLMs can't/are bad at executing - we write programs to help them call tools.
- E.g. calculator, internet search, file management for IDEs like Cursor

Loops

- Need to keep making decisions until reaching the end of the task.
- Initial decision → reassess → next decision

```
while True
        prompt = previous_steps +"if trip is booked return done"
        status = LLM.eval(prompt)
        if status == "done":
                break
```

State management

- The decision space keeps changing - need to remember what was done to know what is next.
- Can be stored in a prompt or external database.

redis

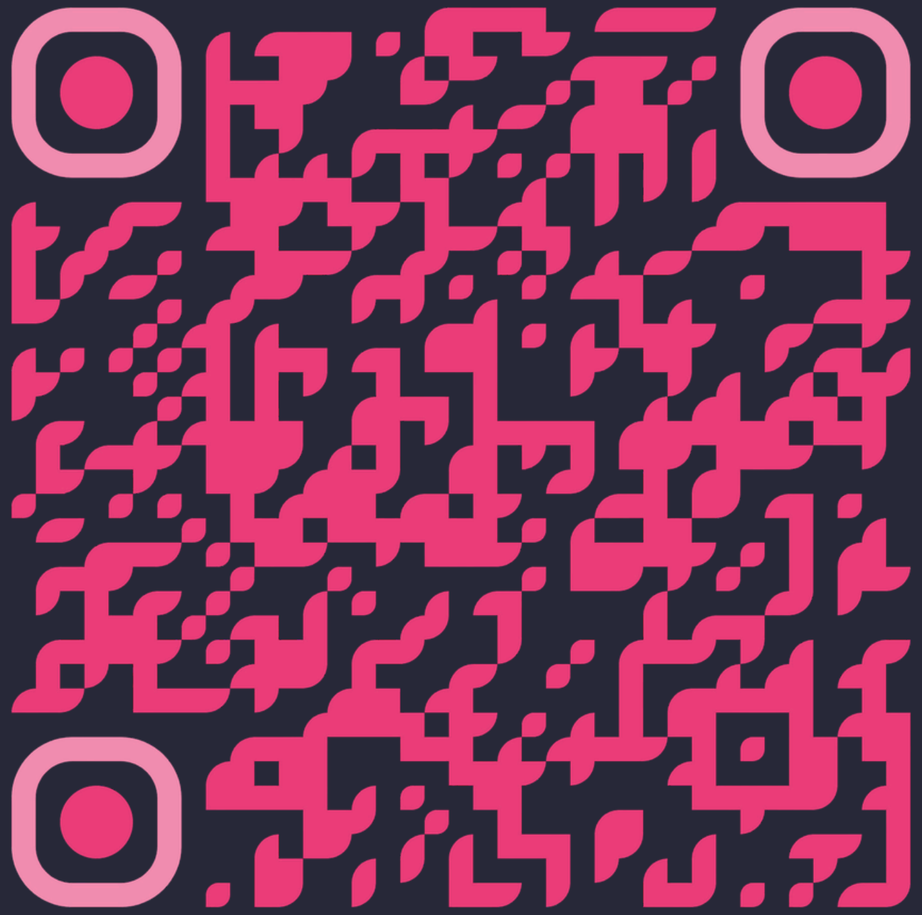
supabase

Shared under Build Club's learning commons. Built for the community. Please share, not sell.

Here’s a demo for a simple agent

Shared under Build Club’s learning commons. Built for the community. Please share, not sell.

Here’s another demo of a simple research agent



Shared under Build Club’s learning commons. Built for the community. Please share, not sell.

Some useful agent-building tools

Workflow builders

zapier

Gumloop

make

Research

LangChain

Multi-agent control platforms

Relevance AI

crewai

Lindy