

OPENCLAW

OpenClaw in 30 Minutes

Get from zero to a working AI assistant that remembers you
— in 30 minutes flat.



Version 1.0 · February 2026 · openclaw-guide.com

□ Table of Contents

01	What You'll Have After This Guide
02	Step 1: Install OpenClaw (5 minutes)
03	Step 2: Set Up Your API Key (3 minutes)
04	Step 3: Start Your First Session (2 minutes)
05	Step 4: Connect Telegram (10 minutes)
06	Step 5: Model Selection Cheat Sheet (5 minutes)
07	Step 6: Common Errors & Fixes (5 minutes)
08	Quick Reference Card
09	You're Done! □
10	Next Steps

Version 1.0 | February 2026

Get from zero to a working AI assistant that remembers you — in 30 minutes flat.

What You'll Have After This Guide

- ❑ OpenClaw installed and running
- ❑ Your first conversation with memory
- ❑ Telegram/Discord connected (talk to it anywhere)
- ❑ Understanding of which model to use for what
- ❑ Common errors and how to fix them fast

Time required: 30 minutes

Prerequisites: A computer, terminal access, \$20 for API credits



Step 1: Install OpenClaw (5 minutes)

macOS / Linux

```
BASH
# Install via npm (recommended)
npm install -g openclaw

# Or use npx for one-off runs
npx openclaw --version
```

Windows

```
BASH
# Using PowerShell
npm install -g openclaw

# Or use WSL2 for best experience
wsl --install
# Then follow Linux instructions
```

Verify Installation

```
BASH
openclaw --version
# Should output: 2026.x.x or similar
```

Stuck? Common fixes:

- ▶ `EACCES` error → Run with `sudo` or fix npm permissions
- ▶ Command not found → Restart terminal, check PATH
- ▶ Node version too old → `nvm install node && nvm use node`

Step 2: Set Up Your API Key (3 minutes)

OpenClaw needs an LLM provider. We recommend **Anthropic Claude** for best results.

Get Your API Key

- 1 Go to console.anthropic.com
- 2 Sign up / log in
- 3 Navigate to API Keys
- 4 Create new key, copy it

Configure OpenClaw

```
BASH
# Create config directory
mkdir -p ~/.openclaw

# Set your API key
export ANTHROPIC_API_KEY="sk-ant-your-key-here"

# Add to shell for persistence
echo 'export ANTHROPIC_API_KEY="sk-ant-your-key-here"' >> ~/.zshrc
# Or ~/.bashrc if using bash
```

Alternative providers:

- ▶ OpenAI: `export OPENAI_API_KEY="sk-..."`
- ▶ Google: `export GOOGLE_API_KEY="..."`
- ▶ Local models: Install Ollama, no key needed

Step 3: Start Your First Session (2 minutes)

BASH

```
# Start a chat session
openclaw chat

# Or specify a model
openclaw chat --model claude-sonnet-4-5
```

You should see:

TERMINAL

```
>Welcome to OpenClaw!
>Type your message or /help for commands.
>
>_
```

Try these first messages:

- 1 "Hi, my name is [your name]. Remember this."
- 2 "What's my name?" (It should remember!)
- 3 "Create a file called hello.txt with 'Hello World' in it"

If all three work, your setup is complete. ☐

Step 4: Connect Telegram (10 minutes)

Talk to your assistant from anywhere.

Create a Telegram Bot

- 1 Open Telegram, search `@BotFather`
- 2 Send `/newbot`
- 3 Name it (e.g., "My OpenClaw Bot")
- 4 Username it (e.g., `my_openclaw_bot`)
- 5 Copy the API token (looks like `123456789:ABCdefGHI...`)

Get Your Chat ID

- 1 Message your new bot
- 2 Go to: `https://api.telegram.org/bot/getUpdates`
- 3 Find `"chat":{"id":123456789}` — that's your Chat ID

Configure OpenClaw

```
BASH
```

```
# Create/edit config
nano ~/.openclaw/config.json
```

Add:

```
JSON
```

```
{
  "channels": {
    "telegram": {
      "botToken": "123456789:ABCdefGHI...",
      "chatId": "123456789"
    }
  }
}
```

```
}  
}
```

Restart and Test

BASH

```
# Restart OpenClaw with Telegram enabled  
openclaw chat --channel telegram
```

```
# Send a message to your bot on Telegram  
# It should respond!
```

Step 5: Model Selection Cheat Sheet (5 minutes)

Don't burn money on the wrong model.

Task	Recommended Model	Why
Main chat	Claude Sonnet 4.5	Best balance of speed/intelligence
Complex reasoning	Claude Opus 4	Deepest analysis, planning
Quick questions	Claude Haiku 3.5	Fast, cheap, good enough
Heartbeats/Crons	Claude Haiku 3.5	No need for Opus on checks
Code generation	Claude Sonnet 4.5	Great at code, reasonable cost
Large context (>100k)	Gemini 1.5 Pro	1M+ context window

Set Default Model

```
BASH
# In config.json
{
  "agents": {
    "defaults": {
      "model": "claude-sonnet-4-5"
    }
  }
}
```

Override Per Session

```
BASH
# Use a different model for this session
openclaw chat --model claude-opus-4

# Or in-message (advanced)
# "Use opus for this: [complex task]"
```

Step 6: Common Errors & Fixes (5 minutes)

"ECONNREFUSED" / Network Errors

```
BASH
# Check internet connection
ping anthropic.com

# Check if behind proxy
export HTTP_PROXY="http://your-proxy:port"
export HTTPS_PROXY="http://your-proxy:port"
```

"Rate limit exceeded"

You're sending too many requests too fast.

```
BASH
# Add delay between requests
# Or upgrade your API tier at console.anthropic.com
```

"Context length exceeded"

Conversation too long. Solutions:

- 1 Start fresh:** Type `/reset` to clear context
- 2 Summarize:** "Summarize our conversation so far, then continue"
- 3 Use larger model:** Switch to Gemini for 1M+ context

Agent Forgetting Things

This is the #1 complaint. Here's the quick fix:

```
BASH
# Create a memory file
```

```
echo "# About Me\n- Name: [Your name]\n- Preferences: [What you like]" > ~/.openclaw/memory.md
```

The agent reads this file on every session.

Want the full memory system? See our advanced playbook →

Quick Reference Card

BASH

Start chat

```
openclaw chat
```

With specific model

```
openclaw chat --model claude-sonnet-4-5
```

With Telegram

```
openclaw chat --channel telegram
```

Reset session

```
/reset
```

Check status

```
/status
```

See all commands

```
/help
```

You're Done!

What you have now:

- ▶ Working OpenClaw installation
 - ▶ API key configured
 - ▶ First conversation with memory
 - ▶ Telegram bot connected
 - ▶ Model selection knowledge
 - ▶ Error troubleshooting basics
-

Next Steps

Want to go further?

☐ **The Autonomous Agent Playbook (\$49)**

- ▶ Full 3-layer memory system
- ▶ Cron + heartbeat mastery
- ▶ Delegation patterns for scale
- ▶ Copy-paste configs
- ▶ Real templates used in production

☐ **Free PDF #2: Cost Optimization**

- ▶ Reduce costs from \$200/mo to \$15/mo
- ▶ Smart model routing
- ▶ Token reduction tricks

☐ **Free PDF #3: Security Checklist**

- ▶ Authenticated vs information channels
- ▶ API key management
- ▶ What to NEVER give your agent

Get all guides at: openclaw-guide.com

Built with ♥ by the OpenClaw community

Questions? Join us at discord.gg/clawd