Sample Configuration

Use this example configuration as a starting point. It includes most keys Codex reads from `config.toml`, along with defaults and short notes.

For explanations and guidance, see:

- Basic Config
- Advanced Config
- Config Reference

Use the snippet below as a reference. Copy only the keys and sections you need into `~/.codex/config.toml`, then adjust values for your setup.

```toml
# Codex example configuration (config.toml)
#
# This file lists all keys Codex reads from config.toml, their default values,
# and concise explanations. Values here mirror the effective defaults compiled
# into the CLI. Adjust as needed.
#
# Notes
# - Root keys must appear before tables in TOML.
# - Optional keys that default to "unset" are shown commented out with notes.
# - MCP servers, profiles, and model providers are examples; remove or edit.

#################################################################################

# Core Model Selection
#################################################################################


# Primary model used by Codex. Default: "gpt-5.2-codex" on all platforms.
model = "gpt-5.2-codex"
```

```
# Model used by the /review feature (code reviews). Default:
        "gpt-5.2-codex".
review_model = "gpt-5.2-codex"

# Provider id selected from [model_providers]. Default: "openai".
model_provider = "openai"

# Default OSS provider for --oss sessions. When unset, Codex prompts.
        Default: unset.
# oss_provider = "ollama"

# Optional manual model metadata. When unset, Codex auto-detects from
        model.
# Uncomment to force values.
# model_context_window = 128000       # tokens; default: auto for
        model
# model_auto_compact_token_limit = 0  # tokens; unset uses model
        defaults
# tool_output_token_limit = 10000     # tokens stored per tool
        output; default: 10000 for gpt-5.2-codex

################################################################################

# Reasoning & Verbosity (Responses API capable models)
################################################################################


# Reasoning effort: minimal | low | medium | high | xhigh (default:
        medium; xhigh on gpt-5.2-codex and gpt-5.2)
model_reasoning_effort = "medium"

# Reasoning summary: auto | concise | detailed | none (default: auto)
model_reasoning_summary = "auto"

# Text verbosity for GPT-5 family (Responses API): low | medium |
        high (default: medium)
model_verbosity = "medium"

# Force-enable reasoning summaries for current model (default: false)
model_supports_reasoning_summaries = false

################################################################################

# Instruction Overrides
################################################################################
```

```
# Additional user instructions are injected before AGENTS.md.
        Default: unset.
# developer_instructions = ""

# (Ignored) Optional legacy base instructions override (prefer
        AGENTS.md). Default: unset.
# instructions = ""

# Inline override for the history compaction prompt. Default: unset.
# compact_prompt = ""

# Override built-in base instructions with a file path. Default:
        unset.
# experimental_instructions_file = "/absolute/or/relative/path/to/
        instructions.txt"

# Load the compact prompt override from a file. Default: unset.
# experimental_compact_prompt_file = "/absolute/or/relative/path/to/
        compact_prompt.txt"


################################################################################

# Notifications
################################################################################


# External notifier program (argv array). When unset: disabled.
# Example: notify = ["notify-send", "Codex"]
notify = [ ]


################################################################################

# Approval & Sandbox
################################################################################


# When to ask for command approval:
# - untrusted: only known-safe read-only commands auto-run; others
        prompt
# - on-failure: auto-run in sandbox; prompt only on failure for
        escalation
# - on-request: model decides when to ask (default)
# - never: never prompt (risky)
approval_policy = "on-request"
```

```
# Filesystem/network sandbox policy for tool calls:
# - read-only (default)
# - workspace-write
# - danger-full-access (no sandbox; extremely risky)
sandbox_mode = "read-only"

################################################################################

# Authentication & Login
################################################################################


# Where to persist CLI login credentials: file (default) | keyring |
        auto
cli_auth_credentials_store = "file"

# Base URL for ChatGPT auth flow (not OpenAI API). Default:
chatgpt_base_url = "https://chatgpt.com/backend-api/"

# Restrict ChatGPT login to a specific workspace id. Default: unset.
# forced_chatgpt_workspace_id = ""

# Force login mechanism when Codex would normally auto-select.
        Default: unset.
# Allowed values: chatgpt | api
# forced_login_method = "chatgpt"

# Preferred store for MCP OAuth credentials: auto (default) | file |
        keyring
mcp_oauth_credentials_store = "auto"

################################################################################

# Project Documentation Controls
################################################################################


# Max bytes from AGENTS.md to embed into first-turn instructions.
        Default: 32768
project_doc_max_bytes = 32768

# Ordered fallbacks when AGENTS.md is missing at a directory level.
        Default: []
project_doc_fallback_filenames = []

# Project root marker filenames used when searching parent
        directories. Default: [".git"]
```

```
# project_root_markers = [".git"]

################################################################################

# History & File Opener
################################################################################


# URI scheme for clickable citations: vscode (default) | vscode-
        insiders | windsurf | cursor | none
file_opener = "vscode"

################################################################################

# UI, Notifications, and Misc
################################################################################


# Suppress internal reasoning events from output. Default: false
hide_agent_reasoning = false

# Show raw reasoning content when available. Default: false
show_raw_agent_reasoning = false

# Disable burst-paste detection in the TUI. Default: false
disable_paste_burst = false

# Track Windows onboarding acknowledgement (Windows only). Default:
        false
windows_wsl_setup_acknowledged = false

# Check for updates on startup. Default: true
check_for_update_on_startup = true

################################################################################

# Profiles (named presets)
################################################################################


# Active profile name. When unset, no profile is applied.
# profile = "default"

################################################################################

# Experimental toggles (legacy; prefer [features])
```

```
################################################################################


experimental_use_unified_exec_tool = false

# Include apply_patch via freeform editing path (affects default tool
        set). Default: false
experimental_use_freeform_apply_patch = false

################################################################################

# Sandbox settings (tables)
################################################################################


# Extra settings used only when sandbox_mode = "workspace-write".
[sandbox_workspace_write]
# Additional writable roots beyond the workspace (cwd). Default: []
writable_roots = []
# Allow outbound network access inside the sandbox. Default: false
network_access = false
# Exclude $TMPDIR from writable roots. Default: false
exclude_tmpdir_env_var = false
# Exclude /tmp from writable roots. Default: false
exclude_slash_tmp = false

################################################################################

# Shell Environment Policy for spawned processes (table)
################################################################################


[shell_environment_policy]
# inherit: all (default) | core | none
inherit = "all"
# Skip default excludes for names containing KEY/SECRET/TOKEN (case-
        insensitive). Default: true
ignore_default_excludes = true
# Case-insensitive glob patterns to remove (e.g., "AWS_*",
        "AZURE_*"). Default: []
exclude = []
# Explicit key/value overrides (always win). Default: {}
set = {}
# Whitelist; if non-empty, keep only matching vars. Default: []
include_only = []
# Experimental: run via user shell profile. Default: false
```

```
experimental_use_profile = false

################################################################################

# History (table)
################################################################################


[history]
# save-all (default) | none
persistence = "save-all"
# Maximum bytes for history file; oldest entries are trimmed when
        exceeded. Example: 5242880
# max_bytes = 0

################################################################################

# UI, Notifications, and Misc (tables)
################################################################################


[tui]
# Desktop notifications from the TUI: boolean or filtered list.
        Default: true
# Examples: false | ["agent-turn-complete", "approval-requested"]
notifications = false

# Enables welcome/status/spinner animations. Default: true
animations = true

# Show onboarding tooltips in the welcome screen. Default: true
show_tooltips = true

# Optional TUI2 scroll tuning (leave unset to use defaults).
# scroll_events_per_tick = 0
# scroll_wheel_lines = 0
# scroll_trackpad_lines = 0
# scroll_trackpad_accel_events = 0
# scroll_trackpad_accel_max = 0
# scroll_mode = "auto"  # auto | wheel | trackpad
# scroll_wheel_tick_detect_max_ms = 0
# scroll_wheel_like_max_duration_ms = 0
# scroll_invert = false

# Control whether users can submit feedback from `/feedback`.
        Default: true
```

```
[feedback]
enabled = true

# In-product notices (mostly set automatically by Codex).
[notice]
# hide_full_access_warning = true
# hide_world_writable_warning = true
# hide_rate_limit_model_nudge = true
# hide_gpt5_1_migration_prompt = true
# "hide_gpt-5.1-codex-max_migration_prompt" = true
# model_migrations = { "gpt-4.1" = "gpt-5.1" }

################################################################################

# Centralized Feature Flags (preferred)
################################################################################


[features]
# Leave this table empty to accept defaults. Set explicit booleans to
        opt in/out.
shell_tool = true
web_search_request = false
unified_exec = false
shell_snapshot = false
apply_patch_freeform = false
exec_policy = true
experimental_windows_sandbox = false
elevated_windows_sandbox = false
remote_compaction = true
remote_models = false
powershell_utf8 = true
tui2 = false

################################################################################

# Define MCP servers under this table. Leave empty to disable.
################################################################################


[mcp_servers]

# --- Example: STDIO transport ---
# [mcp_servers.docs]
# enabled = true                       # optional; default true
# command = "docs-server"                 # required
# args = ["--port", "4000"]               # optional
# env = { "API_KEY" = "value" }           # optional key/value pairs
        copied as-is
# env_vars = ["ANOTHER_SECRET"]            # optional: forward these
        from the parent env
# cwd = "/path/to/server"                 # optional working
        directory override
# startup_timeout_sec = 10.0               # optional; default 10.0
        seconds
# # startup_timeout_ms = 10000              # optional alias for
        startup timeout (milliseconds)
# tool_timeout_sec = 60.0                  # optional; default 60.0
        seconds
# enabled_tools = ["search", "summarize"]  # optional allow-list
# disabled_tools = ["slow-tool"]           # optional deny-list
        (applied after allow-list)

# --- Example: Streamable HTTP transport ---
# [mcp_servers.github]
# enabled = true                          # optional; default true
# url = "https://github-mcp.example.com/mcp"  # required
# bearer_token_env_var = "GITHUB_TOKEN"        # optional;
        Authorization: Bearer <token>
# http_headers = { "X-Example" = "value" }    # optional static
        headers
# env_http_headers = { "X-Auth" = "AUTH_ENV" } # optional headers
        populated from env vars
# startup_timeout_sec = 10.0                   # optional
# tool_timeout_sec = 60.0                      # optional
# enabled_tools = ["list_issues"]             # optional allow-list

################################################################################

# Model Providers (extend/override built-ins)
################################################################################


# Built-ins include:
# - openai (Responses API; requires login or OPENAI_API_KEY via auth
        flow)
# - oss (Chat Completions API; defaults to http://localhost:11434/v1)

[model_providers]

# --- Example: override OpenAI with explicit base URL or headers ---
# [model_providers.openai]
# name = "OpenAI"
# base_url = "https://api.openai.com/v1"         # default if unset
# wire_api = "responses"                          # "responses" |
        "chat" (default varies)
# # requires_openai_auth = true                     # built-in OpenAI
        defaults to true
# # request_max_retries = 4                         # default 4; max
        100
# # stream_max_retries = 5                          # default 5;  max
        100
# # stream_idle_timeout_ms = 300000                 # default 300_000
        (5m)
# # experimental_bearer_token = "sk-example"       # optional dev-only
        direct bearer token
# # http_headers = { "X-Example" = "value" }
# # env_http_headers = { "OpenAI-Organization" =
        "OPENAI_ORGANIZATION", "OpenAI-Project" = "OPENAI_PROJECT" }

# --- Example: Azure (Chat/Responses depending on endpoint) ---
# [model_providers.azure]
# name = "Azure"
# base_url = "https://YOUR_PROJECT_NAME.openai.azure.com/openai"
# wire_api = "responses"                           # or "chat" per
        endpoint
# query_params = { api-version = "2025-04-01-preview" }
# env_key = "AZURE_OPENAI_API_KEY"
# # env_key_instructions = "Set AZURE_OPENAI_API_KEY in your
        environment"

# --- Example: Local OSS (e.g., Ollama-compatible) ---
# [model_providers.ollama]
# name = "Ollama"
# base_url = "http://localhost:11434/v1"
# wire_api = "chat"

################################################################################

# Profiles (named presets)
################################################################################


[profiles]

# [profiles.default]
# model = "gpt-5.2-codex"
# model_provider = "openai"
# approval_policy = "on-request"
# sandbox_mode = "read-only"
# oss_provider = "ollama"
# model_reasoning_effort = "medium"
# model_reasoning_summary = "auto"
# model_verbosity = "medium"
# chatgpt_base_url = "https://chatgpt.com/backend-api/"
# experimental_compact_prompt_file = "./compact_prompt.txt"
# include_apply_patch_tool = false
# experimental_use_unified_exec_tool = false
# experimental_use_freeform_apply_patch = false
# tools_web_search = false
# features = { unified_exec = false }

################################################################################

# Projects (trust levels)
################################################################################


# Mark specific worktrees as trusted or untrusted.
[projects]
# [projects."/absolute/path/to/project"]
# trust_level = "trusted"  # or "untrusted"

################################################################################

# OpenTelemetry (OTEL) - disabled by default
################################################################################


[otel]
# Include user prompt text in logs. Default: false
log_user_prompt = false
# Environment label applied to telemetry. Default: "dev"
environment = "dev"
# Exporter: none (default) | otlp-http | otlp-grpc
exporter = "none"
# Trace exporter: none (default) | otlp-http | otlp-grpc
trace_exporter = "none"

# Example OTLP/HTTP exporter configuration
# [otel.exporter."otlp-http"]
# endpoint = "https://otel.example.com/v1/logs"
# protocol = "binary"                         # "binary" | "json"

# [otel.exporter."otlp-http".headers]
# "x-otlp-api-key" = "${OTLP_TOKEN}"
# Example OTLP/gRPC exporter configuration
# [otel.exporter."otlp-grpc"]
# endpoint = "https://otel.example.com:4317",
# headers = { "x-otlp-meta" = "abc123" }

# Example OTLP exporter with mutual TLS
# [otel.exporter."otlp-http"]
# endpoint = "https://otel.example.com/v1/logs"
# protocol = "binary"

# [otel.exporter."otlp-http".headers]
# "x-otlp-api-key" = "${OTLP_TOKEN}"

# [otel.exporter."otlp-http".tls]
# ca-certificate = "certs/otel-ca.pem"
# client-certificate = "/etc/codex/certs/client.pem"
# client-private-key = "/etc/codex/certs/client-key.pem"
```