

Say This, It Runs That

The complete Flexi Genie AI commands/actions reference — every plain-English request, and the exact action it triggers in your account.

How to use this document

You do not need to memorise any of this. You talk to the assistant normally, in your own words, and it picks the right action.

This document exists for three reasons:

1. **To show you what is possible.** If it is in this list, you can ask for it.
2. **To let you check the assistant's work.** Each action has a name. When the assistant tells you what it did, you can look the name up here and confirm it did the right thing.
3. **To make support faster.** If something goes wrong, telling support "it ran `set_product_pricing` and the price came out wrong" gets you an answer far quicker than "the price is wrong."

Every one of the 176 actions your account exposes is listed here. Nothing is left out.

The one rule that tells you if something is safe

Look at the first word of the action name.

Starts with	What it does	Does it ask you first?
<code>get_</code> , <code>list_</code> , <code>search_</code>	Reads and reports. Changes nothing.	No. It just answers.
Anything else	Changes something in your account.	Yes. Always.

So `get_products` is safe to run any time. `create_product` will always stop, tell you exactly what it is about to create, and wait for you to say yes.

If the assistant ever changes something without asking first, that is a bug. Report it.

*One extra safety layer: for the actions that write page content or publish things live, the assistant runs them ****twice****. The first run only shows you a preview and touches nothing. Only after you approve does it run the second time for real.*

Contents

#	Section	Actions
1	Projects and pages	28
2	The AI page builder	7
3	Funnels and upsell chains	2
4	Products, pricing and payments	25
5	Courses and membership	31
6	Video	8
7	Images and downloadable files	9
8	Email, autoresponders and webhooks	17
9	Sales, customers and subscriptions	13
10	Coupons	7
11	Domains and homepage	6
12	Tracking and Facebook CAPI	9
13	Split tests	8
14	Account and team	6
	Total	176

1. Projects and pages

A **project** is the container. A **page** lives inside it. Almost everything starts here.

Say something like this	It runs
"Show me all my projects."	<code>get_funnels</code>
"What pages are inside my Diwali Sale project?"	<code>get_funnel_pages</code>
"What order are the pages in?"	<code>get_funnel_sequence</code>
"Give me the full picture of this project — pages, products, the lot."	<code>get_funnel_structure</code>
"Which project does this URL belong to?"	<code>search_project_by_url</code>
"What ready-made templates can I start from?"	<code>list_project_templates</code>

"Which of my projects are membership sites?"	list_membership_projects
"Create a new project called Diwali Sale."	create_project
"Make me a copy of the Diwali Sale project."	clone_project
"Take the whole project live."	publish_project
"Move the thank-you page to the end."	reorder_funnel_pages
" Build me a sales page for my yoga course. " <i>This is the main page builder. Use it unless you specifically want the credit-based AI builder in section 2.</i>	create_agentic_page
"Change the headline to 'Join 4,000 students'."	update_agentic_page
"Undo that. Put the previous version back."	rollback_agentic_page
"Show me what's on that page right now."	get_page_html
"What earlier versions of this page do I have?"	get_page_versions
"Change the page title and SEO description."	update_page_settings
"Duplicate my sales page."	duplicate_page
"Publish the sales page."	publish_funnel_page
"Publish the page at this URL."	publish_page_by_url
"Add this chat widget script to the page."	set_page_embed_code
"Show me the standard testimonial block."	get_component_template
"Put a Buy Now button on the page."	add_buy_button_to_page
"Add the checkout form to this page."	add_order_form_to_page
"Add an email opt-in form."	add_lead_form_to_page
"After someone fills the form, send them to the thank-you page."	set_form_redirect
"Who has opted in on this page?"	get_page_subscribers
<i>Runs on its own when a page is too big to send in one go.</i>	request_page_upload_url

2. The AI page builder

This is the credit-based FlexiFunnels AI builder inside your dashboard, and it is separate from the page building in section 1. It uses your AI credits. The assistant will only use it if you name it explicitly — say "use the FlexiFunnels AI builder" and it will.

Say something like this	It runs
"Use the FlexiFunnels AI builder to make me a webinar page."	<code>create_ai_page</code>
"Ask the AI builder to change the colours to green."	<code>customize_ai_page</code>
"Is the AI page finished yet?"	<code>get_ai_page_status</code>
"Show me what the AI builder produced."	<code>get_ai_page_html</code>
"What versions has the AI builder made?"	<code>get_ai_page_versions</code>
"List all my AI-built pages."	<code>list_ai_pages</code>
"Publish the AI page."	<code>publish_ai_page</code>

3. Funnels and upsell chains

A **funnel** here means the money path: what you offer after the first sale. This is not the same as a project.

Say something like this	It runs
"Set up an upsell chain for my course."	<code>create_funnel</code>
"Add a ₹499 order bump as step two."	<code>add_funnel_step</code>

4. Products, pricing and payments

A product is the thing being bought. Pricing sits on top of the product. Payment gateways sit underneath it.

Reading

Say something like this	It runs
"List all my products."	<code>get_products</code>
"Show me the Yoga Masterclass product."	<code>get_product</code>
"How is that product selling?"	<code>get_product_statistics</code>
"What automation rules are on this product?"	<code>get_product_rules</code>
"Which pages can I attach to this product?"	<code>list_pages_for_product_picker</code>
"Which payment gateways does FlexiFunnels support?"	<code>get_payment_gateways</code>
"Which gateways do I actually have connected?"	<code>get_payment_integrations</code>

Creating and editing

Say something like this	It runs
"Create a product called Yoga Masterclass at ₹2,999."	<code>create_product</code>
"Change the product description."	<code>update_product</code>
"Copy this product so I can make a discounted version."	<code>clone_product</code>
"Put this product live."	<code>publish_product</code>
"Take this product off sale."	<code>unpublish_product</code>
"Archive the old 2024 product." <i>Archive is as close to delete as it gets. See section 15.</i>	<code>archive_product</code>
"Attach the checkout page and thank-you page to this product."	<code>set_product_pages</code>

Pricing

Say something like this	It runs
"Set the price to ₹2,999 one-time."	<code>set_product_pricing</code>
"Change the price to ₹3,499."	<code>edit_product_pricing</code>
"Remove the second pricing option."	<code>remove_product_pricing</code>
"Make it ₹999 a month, recurring."	<code>set_product_subscription_pricing</code>
"Change the monthly plan to ₹1,199."	<code>edit_product_subscription_pricing</code>
"Add a ₹499 order bump at checkout."	<code>add_product_bump</code>
"When someone buys this, give them access to the bonus course."	<code>add_product_rule</code>

Payments

Say something like this	It runs
"Connect my Razorpay account." <i>You start this in the dashboard. See section 15.</i>	<code>add_payment_integration</code>
"Turn Razorpay on."	<code>enable_payment_integration</code>
"Turn Stripe off for now."	<code>disable_payment_integration</code>
"Use Razorpay for this particular product."	<code>set_product_payment_integration</code>

5. Courses and membership

The hierarchy is **Course** → **Module** → **Lesson**, and a **bundle** is what people actually buy.

The fast path

Say something like this	It runs
" Build me a sellable course from scratch. " <i>Creates the course, the membership area and the sellable bundle together. Start here.</i>	<code>create_course_funnel</code>

Courses

Say something like this	It runs
"Create a course called Yoga for Beginners."	<code>create_course</code>
"Rename the course."	<code>update_course</code>
"List my courses."	<code>list_courses</code>
"Move Yoga for Beginners to the top."	<code>reorder_courses</code>
"Give me the instructor login link."	<code>get_instructor_login</code>
<i>An old name for creating a course, kept working so nothing breaks. Use <code>create_course</code> instead.</i>	<code>add_course</code>

Modules

Say something like this	It runs
"Add a module called Week One."	<code>add_module</code>
"Rename that module."	<code>update_module</code>
"What modules are in this course?"	<code>list_course_modules</code>
"Release Week Two seven days after purchase."	<code>set_module_drip</code>
"Put this video at the top of the module as an intro."	<code>set_module_introduction_video</code>

Lessons

Say something like this	It runs
"Add a lesson called Breathing Basics."	<code>add_lesson</code>
"Copy this lesson into the next module."	<code>clone_lesson</code>

"Rename the lesson."	update_lesson_title
"Rewrite the lesson description."	update_lesson_description
"What does the lesson description say?"	get_lesson_description
"Unlock this lesson on day 14."	set_lesson_drip
"When does this lesson unlock?"	get_lesson_drip
"Reorder the lessons in this module."	reorder_course_content
"Import this whole YouTube playlist as lessons."	import_playlist_to_module

Lesson attachments

Say something like this	It runs
"Attach the workbook PDF to this lesson."	add_lesson_resource
"What files are attached to this lesson?"	get_lesson_resources
"Make the PDF the main download."	mark_preferred_resource

Selling it

Say something like this	It runs
"Create a bundle called Full Access."	create_bundle
"What bundles do I have?"	list_bundles
"What bundles are in this project?"	list_project_bundles
"Put the Yoga course into the Full Access bundle."	assign_course_to_bundle
"Add just Week One to the free bundle."	assign_module_to_bundle
"Make this product unlock the Full Access bundle."	assign_product_to_bundles
"Show a 'Upgrade to unlock' message on locked lessons."	set_upgrade_gate

6. Video

Four ways to get a video into a lesson. The assistant picks the right one based on where your video is.

Say something like this	It runs
"Use this YouTube / Vimeo link for the lesson."	attach_video_url_to_lesson
"Use this video from another host."	attach_generic_video_to_lesson

"Use the video already in my FlexiFunnels library."	<code>assign_video_to_lesson</code>
"What videos are in my library?"	<code>list_video_gallery</code>
"I want to upload a video file." <i>Two steps: the assistant asks for an upload slot, you upload, then it registers the file.</i>	<code>request_video_upload</code> → <code>register_uploaded_video</code>
<i>The larger-file version of the same two steps.</i>	<code>request_video_upload_url</code> → <code>confirm_video_upload</code>

7. Images and downloadable files

Say something like this	It runs
"Host this image for me."	<code>host_image</code>
"I want to upload an image file."	<code>request_image_upload_url</code>
"What images do I have?"	<code>list_image_gallery</code>
"Put that image in the hero section of my page."	<code>place_image_on_page</code>
"I want to upload a PDF to give away."	<code>request_asset_upload_url</code>
"Set this PDF up as a downloadable freebie."	<code>create_digital_asset</code>
"Show me that digital asset."	<code>get_digital_asset</code>
"List all my downloadable files."	<code>get_digital_assets</code>
"Deliver this PDF when someone fills the opt-in form."	<code>link_asset_to_page_or_form</code>

8. Email, autoresponders and webhooks

Two different wirings live here, and mixing them up is the most common mistake. **An autoresponder connection** links your FlexiFunnels account to a mailing tool. **A form connection** links one specific form to one specific list inside that tool. You need both.

Say something like this	It runs
"Which mailing tools can I connect?"	<code>get_autoresponder_services</code>
"What mailing tools have I connected?"	<code>get_autoresponders</code>
"What lists are in my Mailchimp account?"	<code>get_autoresponder_lists</code>

"Connect my Mailchimp account."	<code>add_autoresponder</code>
"Rename that connection to 'Main Mailchimp'."	<code>rename_autoresponder</code>
"Is this form wired to a list?"	<code>get_form_autoresponders</code>
"Send everyone from this form into my Leads list."	<code>connect_form_to_autoresponder</code>
"Also POST the form data to my own server."	<code>connect_form_to_webhook</code>
"What webhooks are set up?"	<code>get_webhooks</code>
"What are my email settings?"	<code>get_email_settings</code>
"Create an email template for the welcome mail."	<code>create_email_template</code>
"Edit that template."	<code>update_email_template</code>
"Copy the welcome template so I can make a variant."	<code>clone_email_template</code>
"What sending domains do I have?"	<code>get_email_domains</code>
"Add mail.mybrand.com as a sending domain."	<code>add_email_domain</code>
"Verify the sending domain."	<code>verify_email_domain</code>
"What else is connected to my account?"	<code>get_other_integrations</code>

9. Sales, customers and subscriptions

Reading your numbers

Say something like this	It runs
"Show me this month's orders."	<code>get_orders</code>
"Pull up order #4471."	<code>get_order</code>
"List my customers."	<code>get_customers</code>
"Show me everything about this customer."	<code>get_customer</code>
"What sold in the last 24 hours?"	<code>get_recent_sales</code>
"List my active subscriptions."	<code>get_subscriptions</code>
"Show me this subscription."	<code>get_subscription</code>

Changing a subscription — these affect a real customer's billing, so read them twice before saying yes.

Say something like this	It runs
"Pause this customer's subscription."	<code>pause_subscription</code>

"Start their billing again."	resume_subscription
"Cancel this subscription."	cancel_subscription
"Give them another 30 days before the next charge."	extend_subscription_billing
"Email them about their subscription."	send_subscription_notification
" Refund this subscription payment. " <i>Money leaves your account. This cannot be undone from the assistant.</i>	refund_subscription

10. Coupons

Say something like this	It runs
"Create a coupon DIWALI25 for 25% off."	create_coupon
"Show me the DIWALI25 coupon."	get_coupon
"List all my coupons."	get_coupons
"Change DIWALI25 to 30% off."	update_coupon
"Switch that coupon on."	enable_coupon
"Switch that coupon off."	disable_coupon
"Make DIWALI25 work on the Yoga Masterclass only."	assign_coupon_products

11. Domains and homepage

Say something like this	It runs
"What domains do I have?"	get_domains
"Add mybrand.com to my account."	add_domain
"Check whether the domain is pointing correctly."	verify_domain
"Use mybrand.com for this project."	set_project_domain
"What loads when someone visits my root domain?"	get_default_homepage

"Make this page my homepage."	<code>set_default_homepage</code>
-------------------------------	-----------------------------------

12. Tracking and Facebook CAPI

Say something like this	It runs
"What tracking pixels do I have?"	<code>get_capi_pixels</code>
"Add my Facebook pixel."	<code>add_capi_pixel</code>
"Update the pixel's access token."	<code>update_capi_pixel</code>
"Turn tracking on."	<code>enable_capi_pixel</code>
"Turn tracking off."	<code>disable_capi_pixel</code>
"Track purchases of the Yoga Masterclass with this pixel."	<code>link_capi_to_product</code>
"Track the whole Diwali Sale project with this pixel."	<code>link_capi_to_project</code>
"Which pixels are on this product?"	<code>get_product_capi_links</code>
"Which pixels are on this project?"	<code>get_project_capi_links</code>

13. Split tests

Say something like this	It runs
"Split test my two sales page headlines."	<code>create_split_test</code>
"Show me that split test."	<code>get_split_test</code>
"List my running tests."	<code>list_split_tests</code>
"Start the test."	<code>start_split_test</code>
"Pause the test."	<code>pause_split_test</code>
"Rename the test to 'Headline A/B'."	<code>rename_split_test</code>
"How is the test doing?"	<code>get_split_test_report</code>
"Version B won. Send all traffic there."	<code>set_split_test_winner</code>

14. Account and team

Say something like this	It runs
"What plan am I on?"	<code>get_account_settings</code>
"What currency and timezone is my account set to?"	<code>get_account_regional</code>
"Change my currency to INR."	<code>update_account_regional</code>
"Who has access to my account?"	<code>list_subusers</code>
"Invite priya@mybrand.com as a team member."	<code>invite_subuser</code>
"Give Priya access to the Diwali Sale project only."	<code>set_subuser_access</code>

15. What no action will do

There is no delete. Not for pages, not for products, not for courses, not for orders. This is deliberate — a chat window is too easy a place to destroy a live business by accident. The nearest equivalents are `archive_product`, `unpublish_product` and `disable_coupon`, all of which are reversible.

Three things must be started in your FlexiFunnels dashboard, because they involve logging into someone else's service: connecting a payment gateway, connecting a mailing tool, and pointing a domain's DNS. Once you have done the login part, the assistant can take over the rest.

If you ask for something outside this list, the assistant will say so plainly rather than inventing a workaround. That is the correct behaviour.

16. Common sequences

Most real requests are a chain, not a single action. Here are the ones you will hit most. You do not have to say the steps — one sentence is enough and the assistant walks the chain, pausing at each step.

Sell a digital product, start to finish

`create_project` → `create_agentic_page` → `create_product` → `set_product_pricing` →
`set_product_pages` → `add_order_form_to_page` → `set_product_payment_integration` →
`publish_funnel_page` → `publish_product`

Build and sell a course

`create_course_funnel` → `add_module` → `add_lesson` → `attach_video_url_to_lesson` →
`add_lesson_resource` → `create_bundle` → `assign_course_to_bundle` → `assign_product_to_bundles`

Capture leads and deliver a freebie

add_lead_form_to_page → add_autoresponder → connect_form_to_autoresponder →
request_asset_upload_url → create_digital_asset → link_asset_to_page_or_form →
set_form_redirect

Test a headline and keep the winner

duplicate_page → update_agentic_page → create_split_test → start_split_test →
get_split_test_report → set_split_test_winner

Run a discount campaign

create_coupon → assign_coupon_products → enable_coupon → *(after the sale)* → disable_coupon

The short version

- You never need to type an action name. Say what you want in plain English.
- `get_`, `list_` and `search_` read. Everything else changes something and will ask you first.
- Look up the action name the assistant reports back, and you will know exactly what happened.
- Nothing can be deleted. The worst case is almost always reversible.
- `refund_subscription` is the one exception. Money leaves and does not come back.

Every action available in your account is documented above: 176 of them, across 14 areas.