Handling Advanced Data and Resources

- **academy.kspl.tech | Koenig AI Academy**

Prerequisites check

- **You need the Chapter 3 file browser server or an equivalent local MCP server. You should be abl**
- **academy.kspl.tech | Koenig AI Academy**

The resource design problem

- Imagine a support assistant that needs the refund policy in English and German. You could expos
- get_refund_policy(locale)
- That works, but it tells the model "call an action." If the policy is stable context, a resourc
- company-config://policies/refund/en-US
- academy.kspl.tech | Koenig AI Academy

Resource URI rules

- **Good resource URIs are:**
- **Stable: the same policy has the same URI tomorrow.**
- **Meaningful: a human can infer what the resource represents.**
- **Scoped: the URI includes tenant, project, locale, or environment when needed.**
- **academy.kspl.tech | Koenig AI Academy**

Localized configuration example

- Create a config/ directory inside the demo project:
- refund.en-US.json
- refund.de-DE.json
- Each file should contain structured policy data:
- academy.kspl.tech | Koenig AI Academy

Handling binary and large data

- Resources can represent more than text, but large or binary data needs care. If a PDF contract
- Expose metadata first: title, type, page count, owner, updated time.
- Expose section resources: contract://123/section/payment-terms.
- Expose summaries with provenance: include page numbers or section IDs.
- academy.kspl.tech | Koenig AI Academy

Try it next

- **Deploy a gateway with RBAC, rate limits, and JSONL auditing**
- **academy.kspl.tech | Koenig AI Academy**