Perfect 👌 — that’s the right approach: know the **concepts**, anticipate **interview questions**, and also **showcase a mini-project** on GitHub that demonstrates your understanding of **integrations + domain separation**.

**🔹 1. Interview-Style Questions**

Here’s what you might get asked:

**Conceptual Questions**

1. **What is Domain Separation in ServiceNow?**
   * Explain how it allows multiple organizations to share a single instance but maintain data isolation.
2. **How does domain separation affect integrations?**
   * Mention sys\_domain field, domain-specific configurations, and security implications.
3. **If an external system creates incidents in ServiceNow, how do you ensure the record lands in the correct domain?**
   * Answer: By setting the sys\_domain field explicitly in the payload or via a domain-separation transform map/business rule.
4. **How do you handle multi-tenant integrations with a single external system?**
   * Example: One monitoring tool feeds multiple customer domains → Use routing logic (like a domain lookup table or integration middleware) to map source identifiers to the correct sys\_domain.
5. **What are challenges when integrating with a domain-separated instance?**
   * Data leakage, debugging integrations across domains, global vs. domain-specific configurations, and performance.

Got it 👍 — since you’ve now decided to leverage **core\_company** (instead of u\_domain\_mapping), let’s update the **Steps** and **Progress Chart** so your GitHub and demo documentation look sharp.

**🔹 1. Steps for Domain-Separated Integration (Using core\_company)**

**Phase 1 – Setup Domains**

1. Installed **Domain Support – Domain Extensions** plugin.
2. Verified **Domain [domain]** table exists with OOB entries (ACME, Cisco, Initech, etc.).
3. Created two new test domains:
   * DomainA (Customer, Parent = TOP)
   * DomainB (Customer, Parent = TOP)

**Phase 2 – Setup Companies**

1. Navigated to **Companies [core\_company]**.
2. Created test company records:
   * CompanyA → assigned to DomainA
   * CompanyB → assigned to DomainB

**Phase 3 – Setup Users & Visibility**

1. Created two test users:
   * userA → Domain = DomainA
   * userB → Domain = DomainB
   * Assigned role itil.
2. Configured **User Visibility Domains** so:
   * userA can only see DomainA incidents.
   * userB can only see DomainB incidents.

**Phase 4 – Scripted REST API**

1. Built a **Scoped App** with a Scripted REST API:
   * Accepts JSON payload with company and short\_description.
   * Looks up core\_company by name.
   * Uses the company’s domain (core\_company.domain) to set incident.sys\_domain.
   * Creates an Incident with proper Company + Domain assignment.

**Phase 5 – Testing**

1. Tested via **Postman**:
   * Sent request for CompanyA → Incident created in DomainA.
   * Sent request for CompanyB → Incident created in DomainB.
2. Verified in UI:

* Logged in as userA → only sees DomainA incidents.
* Logged in as userB → only sees DomainB incidents.

**🔹 2. Progress Chart (Spreadsheet-Style)**

| **Phase** | **Task** | **Status** |
| --- | --- | --- |
| Setup | Install Domain Extensions plugin | ✅ Done |
| Setup | Verify Domain table | ✅ Done |
| Setup | Create test domains (DomainA, DomainB) | ✅ Done |
| Company | Create companies (CompanyA, CompanyB) | ✅ Done |
| Company | Assign domains to companies | ✅ Done |
| Users | Create test users (userA, userB) | ✅ Done |
| Users | Assign users to domains | ✅ Done |
| Users | Configure User Visibility Domain rules | ✅ Done |
| API | Create Scoped App & Scripted REST API | 🔄 In Progress |
| API | Query core\_company for domain lookup | 🔄 In Progress |
| Test | Postman test: CompanyA → DomainA incident | ⏳ Pending |
| Test | Postman test: CompanyB → DomainB incident | ⏳ Pending |
| Test | Verify domain-restricted visibility in UI | ⏳ Pending |
| Deliverable | Document steps in GitHub README | ⏳ Pending |

✅ This way you have a **clear checklist for yourself** and a **professional progress log** for GitHub/interview showcase.

👉 Do you want me to now **drop in the final Scripted REST API code (using core\_company + sys\_domain)** so you can complete the “API” row in your progress chart?