Here's a roadmap tailored for your preparation for interviews based on your background and goals:

---

1. Technical Skills Review

Core Python

- Revise Python fundamentals: data types, loops, conditionals, comprehensions.
- Advanced topics: decorators, generators, context managers, multithreading, and multiprocessing.

Django Framework

- Core concepts: Models, Views, Templates, Middleware.
- ORM: Complex queries, relationships (OneToOne, ForeignKey, ManyToMany).
- Django REST Framework (DRF): Serializers, ViewSets, authentication, and permissions.
- Deployment: Django on production servers (Gunicorn, Nginx).

Database Skills

- PostgreSQL: Write and optimize complex queries, indexing, and transactions.
- ORMs: Efficient query optimization.
- Work with database migrations.

Asynchronous Services

- Kafka and RabbitMQ: Understand their architecture, differences, and use cases.
- Celery workers: Task queues, result backend, periodic tasks, and best practices.

API Development

- RESTful principles: CRUD operations, versioning, status codes, and documentation.
- Security: Authentication, authorization, rate-limiting, and data validation.

DevOps and Deployment

- Docker: Create and manage containers for your apps.
- CI/CD: Jenkins, GitHub Actions, or similar tools.
- Cloud: Familiarize yourself with AWS or GCP basics.

---

2. System Design

- Study design patterns: Singleton, Factory, Observer, etc.
- Scalable system design: Learn to design microservices, databases, caching, and load balancers.
- Common scenarios: URL shortening, chat applications, or payment gateways.
- Resources: "System Design Primer" on GitHub, YouTube tutorials, and case studies.

---

3. Coding Interviews

Data Structures and Algorithms

- Focus Areas:
  - Arrays, Strings, Linked Lists, Stacks, Queues, HashMaps.
  - Trees, Graphs, Dynamic Programming, and Backtracking.
- Practice on platforms like LeetCode, HackerRank, or CodeSignal.
- Specifically, solve problems involving iteration counts in loops (your challenge area).

---

4. Behavioral Interviews

- Prepare STAR (Situation, Task, Action, Result) stories for:
  - Challenges you’ve faced.
  - Achievements in your projects.
  - Instances of team collaboration and conflict resolution.
- Common questions:
  - "Why do you want to switch roles?"
  - "Describe a challenging problem you solved in your current role."

---

5. Mock Interviews

- Conduct mock interviews with peers or platforms like Pramp.
- Request feedback on technical, problem-solving, and communication skills.

---

6. Networking

- Update your LinkedIn profile to highlight your skills and experience.
- Reach out to former colleagues and mentors for referrals and insights.

---

7. Java and Spring Boot

Since you're planning to transition into Java and Spring Boot:

- Familiarize yourself with Java basics and the Collection Framework.
- Begin learning Spring Boot for REST API development and microservices.
- Implement small projects to demonstrate proficiency.

Resources

- **Books**:
  - *Cracking the Coding Interview* by Gayle Laakmann McDowell.
  - *Designing Data-Intensive Applications* by Martin Kleppmann.
- **Courses**:
  - "Django for Professionals" by William S. Vincent.
  - System design courses on Udemy or Educative.
- **YouTube Channels**: Tech Dummies, Gaurav Sen for system design.

Would you like assistance with any specific section or resources for these topics?