Java Interview Cheat Sheet – JPMorgan Chase (Core Java Focus)

Core Java – OOPs, Language Features

- What is the difference between '==' and '.equals()' in Java?
- What are the principles of OOP and how is each implemented in Java?
- Explain immutability in Java. How do you create an immutable class?
- What is the difference between 'final', 'finally', and 'finalize()'?
- How does Java achieve platform independence?

Collections & Generics

- Difference between ArrayList and LinkedList? When to use which?
- What is the difference between HashMap, TreeMap, and LinkedHashMap?
- Why are HashMap keys usually immutable?
- What are fail-fast and fail-safe iterators?
- Explain how ConcurrentHashMap works.

Multithreading & Concurrency

- What is the difference between synchronized block and method?
- How does ReentrantLock differ from synchronized?
- Explain thread lifecycle in Java.
- What is a deadlock? How do you prevent it?
- What is the use of volatile keyword?
- Explain the Executor Framework. What is the difference between Callable and Runnable?
- What is the difference between wait() and sleep()?

JVM Internals & Memory Management

- What are the different memory areas allocated by JVM?
- Explain garbage collection and types of GC algorithms in Java.
- What is a memory leak in Java?
- How can you profile memory usage in a Java application?

Exception Handling

- What is the difference between checked and unchecked exceptions?
- Best practices for exception handling in large-scale applications?
- Can a finally block suppress an exception?

Java 8+ Features

- What is a lambda expression?
- Explain method references and functional interfaces.
- What is Stream API? How does lazy evaluation work in streams?
- Difference between map() and flatMap()?
- What are Optionals and when should you use them?

Design & Best Practices

- Explain SOLID principles with examples.
- Have you used any design patterns? Describe a few.
- How do you ensure thread safety in a shared-resource environment?
- How would you design a rate limiter or a cache system?

Behavioral & JPMorgan-Specific

- Tell me about a time you worked under pressure on a deadline.
- How do you handle legacy systems and modernize Java code?
- Have you worked in finance or risk systems? Explain your role.
- Why JPMorgan?
- What is your understanding of Agile/DevOps in a financial organization?