Resources used for building weatherApp project. (18/02/2023)

1. Runnable & Callable interfaces:
   https://www.baeldung.com/java-runnable-callable#:~:text=The%20Runnable%20interface%20is%20a,%7B%20public%20void%20run()%3B%20%7D

2. ExecutorService interface: https://www.baeldung.com/java-executor-service-tutorial

3. Future & CompletableFuture objects:
   https://www.javatpoint.com/java-future-example ,
   https://www.javatpoint.com/completablefuture-in-java#:~:text=A%20CompletableFuture%20is%20an%20extension,the%20computation%20when%20it%20completes

4. @Value annotation: https://www.baeldung.com/spring-value-annotation

5. Function interface:
   https://www.geeksforgeeks.org/function-interface-in-java-with-examples/#:~:text=The%20Function%20Interface%20is%20a,argument%20and%20produces%20a%20result

6. Unmodifiable Map & Immutable Map:
   https://www.baeldung.com/java-immutable-maps

7. @ConfigurationProperties & @EnableConfigurationProperties annotations:
   https://zetcode.com/springboot/configurationproperties/#:~:text=The%20%40Configuration%20annotation%20makes%20it,the%20prefix%20for%20our%20properties.&text=In%20the%20MyRunner%20%2C%20we%20inject,field%20and%20read%20its%20values.&text=Application%20is%20the%20entry%20point%20which%20sets%20up%20Spring%20Boot%20application ,
   https://www.baeldung.com/configuration-properties-in-spring-boot

8. AtomicReference:
   https://www.tutorialspoint.com/java_concurrency/concurrency_atomicreference.htm#:~:text=AtomicReference%20class%20provides%20operations%20on,on%20underlying%20object%20reference%20variable.

9. Chain of Responsibility principle:
   https://www.digitalocean.com/community/tutorials/chain-of-responsibility-design-pattern-in-java

10. Interceptors in OkHttp Java:
    https://www.baeldung.com/java-okhttp-interceptors ,
    https://square.github.io/okhttp/features/interceptors/

11. Correlation ID:
    https://www.oreilly.com/library/view/building-microservices-with/9781785887833/1bebcf55-05bb-44a1-a4e5-f9733b8edfe3.xhtml ,
    https://www.okta.com/identity-101/correlation-id/ ,
    https://labs.leaseweb.com/using-correlation-ids-in-api-calls-42b1cd84813b

12. MDC (Mapping Diagnostic Context):
    https://medium.com/@eranda/mapped-diagnostic-context-mdc-b6c81ff426f1 ,
    https://howtodoinjava.com/logback/logging-with-mapped-diagnostic-context/
    !) **Note:** Correlation ID and MDC are used in combination with each other!

13. Thread Pools in Java:
    https://www.geeksforgeeks.org/thread-pools-java/
    https://www.baeldung.com/thread-pool-java-and-guava

14. Factory Design Pattern: https://www.digitalocean.com/community/tutorials/factory-design-pattern-in-java

15. ThreadFactory interface: https://www.geeksforgeeks.org/threadfactory-interface-in-java-with-examples/

16. Flyway migration in Spring Boot: https://www.baeldung.com/database-migrations-with-flyway , https://flywaydb.org/documentation/concepts/migrations, https://blog.devgenius.io/database-migration-in-spring-boot-using-flyway-ee791db8aea0, https://www.youtube.com/watch?v=ovG1wgEqE10 , https://www.youtube.com/watch?v=ltdq3vl5g_U, https://www.section.io/engineering-education/spring-boot-flyway-migrations/, https://medium.com/capital-one-tech/database-migrations-with-flyway-16b6478e55a6, https://www.youtube.com/watch?v=Wo0gXUVjy2A, https://dzone.com/articles/database-versioning-with-flyway-and-java, https://dev.to/trustbirungi/getting-started-with-flyway-in-spring-boot-po1

17. Flyway baseline migration profiles in application.yml : https://flywaydb.org/documentation/concepts/baselinemigrations , https://flywaydb.org/documentation/tutorials/baselineMigrations , https://flywaydb.org/documentation/configuration/parameters/outOfOrder

18. Database Cache: https://stackoverflow.com/questions/12227752/what-is-a-database-cache-and-how-does-one-use-it, https://www.quora.com/What-is-database-cache

19. NamedParameterJdbcTemplate class: https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/jdbc/core/namedparam/NamedParameterJdbcTemplate.html

20. How to push submodules of project into GitHub repository (through Git): https://www.youtube.com/watch?v=gSlXo2iLBro

21. How to create custom deserializer in java: https://www.baeldung.com/jackson-vs-gson https://www.baeldung.com/jackson-custom-serialization https://www.baeldung.com/jackson-deserialization https://www.baeldung.com/java-jsonnode-get-keys

22. How does JsonNode work and how to parse Json object from Json array: https://mkyong.com/java/jackson-tree-model-example/

23. What is Batch Operation in JDBC and how to use it: https://www.baeldung.com/jdbc-batch-processing https://docs.spring.io/spring-framework/docs/3.0.0.M3/reference/html/ch13s04.html

24. TaskScheduler and Dynamic Scheduling: https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/scheduling/annotation/SchedulingConfigurer.html https://samuel-mumo.medium.com/one-stop-guide-for-implementing-scheduling-in-spring-boot-660dec88a25e

https://learnwithtaufik.home.blog/2019/03/05/dynamic-scheduling-with-spring-boot/
https://www.baeldung.com/spring-task-scheduler
https://docs.spring.io/spring-framework/docs/4.2.x/spring-framework-reference/html/scheduling.html#:~:text=The%20TaskExecutor%20was%20originally%20created,TaskExecutor%20abstraction%20to%20pool%20threads.

25.	Making methods Async:
https://spring.io/guides/gs/async-method/

26.	Event-driven programming:
https://www.callicoder.com/java-8-completablefuture-tutorial/
https://stackoverflow.com/questions/35809827/java-8-completablefuture-allof-with-collection-or-list
https://kalpads.medium.com/fantastic-completablefuture-allof-and-how-to-handle-errors-27e8a97144a0