

Cloud

17 July 2024 11:59

Things required to build an App.

Code Editor
Frontend
Backend
DataBase
API's

1. Types of servers and configurations.
2. How software is converted to Image.
3. what is enterprise service bus

MICROSERVICES

LIMITATIONS OF MONOLITH

- Difficult to integrate
- Scalability
- Scaling is of two types : Horizontal and Vertical scaling
- Not easy to upgrade
- SOA - Service Oriented Architecture

-
- Interoperability

-
- SOA principles are realised and implements using microservices
-

Store

1. Eggs
 2. Milk
 3. Chocolates
 4. Apples
 5. Bread
-
1. Eggs in ArrayList
 2. Milk in HashSet
 3. Chocolates Linked list
 4. Apples on TreeSet
 5. Bread on Queue

Queries

1. What is the availability of milk/eggs
2. What is the total cost of milk today
3. What is the total cost of all products

Using log without sysout

Xuvpi9-robvaz-diwnyp

- Post Construct
- Pre Destroy
- Component
- Service
- Configuration
- Primary
- Qualifiers

- Bean
- Controller
- ComponentScan
- Property
- Value

1. @Component is employed to designate general-purpose Spring components.
2. @Service is utilized for managing service layer beans in Spring applications.
3. @Controller is employed for defining Spring MVC controllers.
4. @Configuration is used to specify Spring configuration classes.
5. @Bean is utilized within @Configuration to define individual beans in Spring.
6. @ComponentScan is employed to instruct Spring on which packages to scan for components.

- ▶ @Import- for importing classes
- ▶ @Profile - Dev Test Prod
- ▶ @ImportResource({"spring-context.xml"})

SPRING BOOT

- i. Spring-Configuration +(Automation + Starters + Actuators+ DevTools + Cloud Tools) = SpringBoot
- ii. Spring boot = Ten years of spring experience
- iii. Spring boot = Agile development + dependencies starters + inbuilt tomcat server
- iv. Spring boot - Microservices -> Restful API
- v. @RestController = Controller + ResponseBody -<body> , http headers others get you will have body, (pay laod)
- vi. Controller -> commit or render views using view technologies(HTML or JSP or Tiles or Velocity....)

1. @PostConstruct
2. @PreDestroy
3. @Qualifier
4. @Primary
5. @Component
6. @Service
7. @Configuration
8. @Controller
9. @ComponentScan
10. @RestController
11. @RequestMapping
12. @GetMapping
13. @PostMapping
14. @PathVariable
15. @RequestBody
16. @RequestHeader
17. @Value
18. @Autowired
19. @Lombok

- mvn spring-boot : run
- .\mvnw spring-boot : run

CURL - TODO WORK

GARBAGE COLLECTOR JVM

- API GATEWAY IS ALSO KNOWN AS EDGE SERVICE.
- It also performs load balancing and also applies security.

25/07/2024

1. Cloud GateWay - Ports - Registry : 8761, HelloService : 8071 , HelloClient : 8072 , Gateway : 8070
2. Cloud Config - Server port : 88888
3. OpenFiegn - <https://spring.io/projects/spring-cloud-openfeign> , <http://httpbin.org>
4. Circuit Breakers -

- @EnableConfigure
- @FeignClient
- @EnableDiscoveryClient

26/07/2024

@Autowired
Environment environment

```
environment.getProperty("server.port");  
// port no will be displayed
```

1. Circuit Breaker Introduction
2. Gateway load balancing
3. Sleuth

LOAD BALANCING

Cloud:gateway:discovery:locator:enabled:true -to enable auto load balancing
http:localhost:8761/eureka
Services - lbeureka , lbsevice, lbgateway

SLEUTH

[application name, traceId, spanId, export]

- Application name – This is the name we set in the properties file, and can be used to aggregate logs from multiple instances of the same application.
- TraceId – This is an id that's assigned to a single request, job, or action. Something like each unique user initiated web request will have its own traceId.
- SpanId – Tracks a unit of work. Think of a request that consists of multiple steps. Each step could have its own spanId and be tracked individually. By default, any application flow will start with the same TraceId and SpanId.
- Export – This property is a boolean that indicates whether or not this log was exported to an aggregator like Zipkin. Zipkin is beyond the scope of this article, but plays an important role in analyzing logs created by Sleuth.

1. @EnableMethodSecurity
2. @CircuitBreaker
3. @Sleuth

-
1. JDBC Authentication

-
- Circuit Breakers
 - Docker (Kubernetes - Messaging with Rabbit MQ and kafka) - Load Balances + circuit breaker
 - Migrate your store project to Spring boot - microservices

- Docker steps :
 - 0. Create a new Folder and switch to that folder.
 - 1. Create Docker File.
 - 2. Define the libs/bins.
 - 3. Build an image with a docker file.
 - 4. Run the image file to instantiate the container - run time instance of an image which is based on Docker file specification.
-

DOCKER COMMANDS:

- Docker build . // builds Dockerfile
- Docker images // shows docker images
- Docker build -t name:id // naming while building

- Docker run -t imageid / name:id
- Docker -d -p 9000:80 iamgeid // change

- Docker ps -a
- Docker run -it imageid bash
- Docker stop imageid [new cmd]

Kubernetes

Kubectl exec -it service/nginx-svc -- bash [to enter into the running container on pod]

AMAZON AWS

- EC2
- S3
- RDS
- Integration - Connecting Ec2 with RDS
- Elastic Bean Stalk
- Code Commit

Loose Coupling

- Loose coupling is when objects doesn't depend on each other but depends on abstraction.
- As there is minimal dependency in the classes , changes of one class will not affect the other class.

FEEDBACK

- How confidently you said name
 - Eye contact
 - Topic - difficulty/contact
 - Demonstrate
 - Time Management
-

CodeBuild

PENDING
DONE
IN PROGRESS