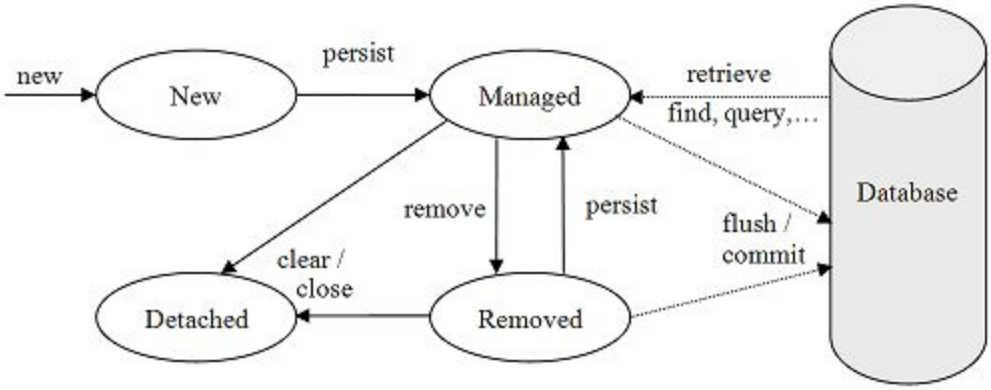
what is collection
what is a framework
what is a set, list and queue
how do sets work? How does it not add any duplicate elements?
ArrayList vs array
Vector vs array
LinkedList vs ArrayList
what is default size of ArrayList? (answer=0)
what are the three ways of calling a method from other class which is not a super or sub class? (ans: third way: forName();)
what are the ways of creating any object in java? (Answer: use new keyword, use forName(), use clone or static factory.. these use new keyword
internally, use deserialization)
throws vs throw
custom exception runtime vs compile time
how do we call a static class(which is inside another class) to use Comparator ?
By default what is the class type of the object returned via deserialization? (Ans: java.lang.Object)
Access Modifiers = access specifiers(private public protected) + something else(static, final, abstract, synchronised, transent, volatile, strictfp)
what does transient access modifier do? -> prevents the data from being serialized. default value is returned/used.
who manages all the threads in java? Thread scheduler : allocates task, buffer space / resource and priority
What exception do we get when a thread is disturbed in sleep mode? InterruptedException
What is ThreadGroup?
What is the maximum capacity of a thread pool? How does it work
What exception appears on passing an illegal priority in thread.setPriority? IllegalArgumentException
Will a higher priority thread ALWAYS execute before one with low priority? No, it's upto the scheduler. Priority is only a recommendation to TS
What exception do you get when you try to start a thread which is already started? IllegalThreadStateException
Why does JDBC have all interfaces? :So that the vendors of different databases can implement them according to the type of database!
When do we get IllegalMonitorStateException?
Why do we use .properties file? To facilitate internationalization or independency.
By default, resultsets are forward only and read only.
what is concurrent hashmap?
enum vs static final
sleep vs wait
how do wait(), notify() and notifyAll() work?
difference between statement, preparedStatement and callable

what are the two rules for final variable? It should always be initialised and it should never be reassigned.
why do we use BigDecimal(String) or BigDecimal(Long)
how to start a transaction in java? connection.autoCommit(false) and set connection.rollback in catch block, connection.commit() commits the queries
and automatically sets autoCommit to true
preparedStatement vs Statement -> Query gets precompiled when you call connection.prepareStatement(sql) method and AFTER that, the values for ?
is passed to the precompiled query. To understand better, check the color of the ps object in eclipse. It is blue -> static -> compile time creation
What are loggers and why?
What are different log levels?
What are appenders?
What are layouts in Logger
What are patterns in Logger
Can you throw an exception in a static block? No.
When do we get ForeignKeyConstraintViolated Exception?
what is sonarqube, why do we use it? :continuous monitoring of code tool
In how many ways can we get the logger details? : one is specifyign the .class fle name and another is by using getClass();
Why do we create reference of super class, while creating object of subclass? Demonstrate how references work in inheritance

JPA

What is JPA? Why do we need it?
What is ORM? Explain
what is persistance.xml
what are the different tools of ORM? what is the role of each tool? hibernate top link etc
JPA Entity Life cycle?



Explain different annotations in JPA
Difference between EntityManagerFactory and EntityManager?
CRUD Operations
What is JPQL(Java Persistence Query Language)? Explain Dynamic Query, Named Query and Typed Query, Native Query and Stored Procedure. When do we use them
Entity association?
What is cascading? Explain all types of cascading and their roles
Explain the attributes inverse column, attributes, mapsby, Join column
What is eager loading and lazy loading?
Inheritance in JPA: Explain the strategies
How does ORM help solving the 4 problems in JDBC namely Granularity, Identity, Association and Inheritance
what is a dialect?
What is hbm2ddl.auto property and what are the values it can take?(Ans: create, update, create-drop, drop). Explain those values :: create-no need to create tables, once created, change to update to use the same database.

NOTE: JPA cannot map the legacy java class “Date” by itself, it has to be explicitly specified under @Temporal(TemporalType.TIMESTAMP or DATE)

Explain @Transient annotation
Is joincolumn always necessary?
What is Target?
Explain mappedBy

How do you map a non primary key using @JoinColumn? : use referencedColumnName attribute inside @JoinColumn and mark the nonprimary key column as @NaturalId

SPRING CORE

What is spring? Why do we use spring DI(Dependency injection) and IOC?
Which design patterns do we use in spring?
Explain factory design pattern and spring container. Also explain the difference
How can we create bean autowiring? Explain all the ways

NOTE: Spring is an open source, lightweight DI and AOP container based framework. Spring provides Aspect and Cross Cutting concerns. It offers Advises: @Before, @After, @Around, @Afterthrowing @AfterReturning
Spring supports 2 DIs, Constructor and Setter(or Property)

What are the problems solved by Spring? : Testability, Maintainability, Scalability, Complexity, helps the user focus on business logic, without having to worry about the framework we use
Spring container (lifecycle, difference of BeanFactory and ApplicationContext)
How to create bean? Explain all the different ways
Explain the annotations
Difference between pattern and framework: Pattern: rules

What is setter injection?

NOTE: invalid setter method exception : Occurs when you initialize a property in spring.xml without having a setter method in the class.

Note: <bean id=””> , here id is the name of the object created (the string id spring searches for)

What will be the values set in the property of a bean when you are using constructor autowire but are also defining the property values explicitly in the <property> tag?: Ans: the property tag value will remain because setters are always called after the constructors so that will overwrite the constructor initializations

What are the different annotations used to create beans in different layers? : @Component for UI, @Service for service layer and @Repository for DAO layer. @Bean for any other layer.

What are the bean scopes in spring?

SPRING MVC

What is mvc? Difference between applications
Spring MVC flow
ApplicationServer and WebServer

What is Web container?
Front controller
What is EE?
View Resolver?
Model and view difference
@Controller, @RequestMapping, @ModelAttributes, @RequestParam, @Validate, @RestController @pathvariable
How can we do server side validation?

JSP
What is JSP? Why do we use it?
Life cycle of a JSP?
What are directives? (page, taglib and import)
What are JSP implicit objects?
include, forward
RequestDispatcher, SendRedirect
Servlet config and context : (config is specific to a particular servlet, servlet context is global)
Explain the errors 404, 500, 402, 403, 503, 200
Difference between GET, POST, PUT and DELETE
When is a dialect configured or created?

SPRING WITH JPA

Explain the flow.. How do we integrate Spring MVC with JPA
What is: @PersistanceContext, <tx:annotation-driver> , JPATransactionManager, LocalContainerEntityManagerFactory, @Transactional

modalAttribute: fetches object from controller to form
@ModalAttribute: fetches object from view to controller

WEB SERVICES

What are web services?
Difference between web service and web application:
Web service-used by another applications(code) **WebApplication**-used by end users
Why do we need/use them
What are SOAP and REST? Explain the difference
WSDL, UDDI
@RequestParam, @ModelAttribute, @RequestBody, @ResponseBody

SPRING BOOT

What is spring boot, why are we using spring boot? : basic spring mvc is slow, everythng passes through the front controller having all the configurations which makes it slow
Flow of spring boot application
How can we add new server in spring boot?
How can we integrate spring MVC, JPA in Spring Boot
Difference between @RestController and @Controller
Difference between @GetMapping, @PostMapping, @PutMapping, @DeleteMapping and @RequestMapping

SPRING DATA

What is Spring Data? Why do we need/use it?
Flow of Spring Data
Difference between CRUDRepository, JPARepository, PaginationRepository

SPRING SECURITY

What is Spring Security? Why do we use it?
What are the 5 key points to remember in Spring Security?
5 key points to remember : Authentication, Authorization, Principle, Grant Authority and Roles

Principle : security only understands accounts, not people. Meaning: Same user can have multiple accounts but for the system/security, all of those accounts are separate and unique
Grant Authority Table: Takes care of permissions for each URL
AuthenticationManagerBuilder : the “guard”

What is the flow of Spring security? : Start by explaining what each dependency does in pom.xml (Example: starter-security creates the “guard”)
Difference between AuthenticationManagerBuilder, WebSecurity, HttpSecurity

JWT

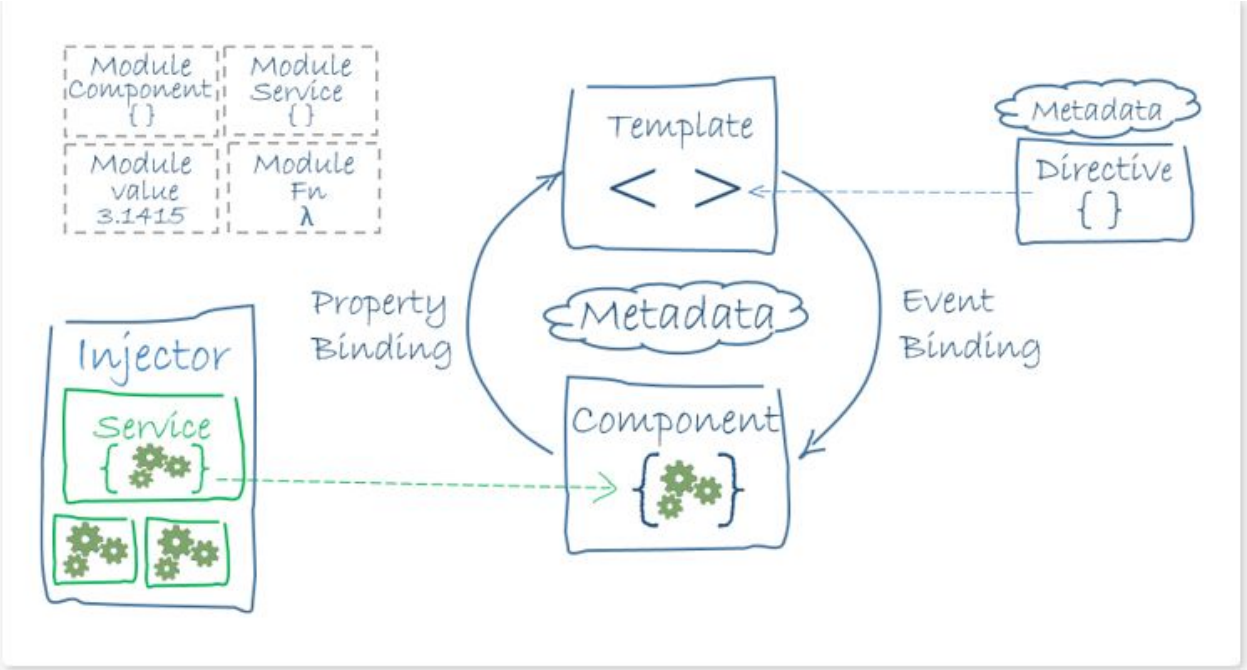
What is JWT(JSON based WEB TOKENs) and why do we use it?
3 parts of a JWT Token? : Header, payload, signature

ANGULAR 6

What and why: Angular : UI framework which enables us to create a **single application** for **all platforms**(web, mobile and desktop application)
Why Angular :

1. Open Source
2. Client Side
3. UI Framework
4. SPA

What is SPA? (Single Page Application) : <A single-page application is an app that works inside a browser and does not require page reloading during use- Src : Medium>
Versions: Current: v8.2.11, We are using: v6
What are **Building blocks** of Angular?
A: Building Blocks of Angular Application(Architecture):



1. Injector/Service: Make contract between Component and Backend
2. Component: It consists of metadata and templates.(One application has many components)
    a. Metadata: property config of the component, like selector, templateUrl
    b. Template: View or the HTML
3. Directive: Used to attach behaviour elements in DOM (Anything apart from static HTML content comes under Directive, including the components)
   Three kinds of Directives:
    - **Components**: directives with template

- **Structural directives:** Change the DOM layout by adding or deleting DOM elements(predefined)
    - *ngId
    - *ngFor
    - *ngSwitch
- **Attribute directives**: Change the appearance or behaviour of an element, component or another directive

Alternate technologies: REACT, Vue, meteor etc

********** Src : STACK OVERFLOW **********

- Tomcat is a **web server** for **web applications** written in **java**.
- Node is a **runtime environment** for **applications** in **javascript**.

`JVM vs. V8+Node,`

`Java vs. Javascript,`

`Tomcat vs. Express`

********** P.S. - Express is a popular Node Server **********

What and why : **Node.js**

- Its an open source **server** framework
- It helps us to run javascript on server -> main function
- Create Dynamic page content
- It can create open read write delete and close files on the server

What is NPM ?- **Node Package Manager** : implicitly download all the dependencies required (similar to starter in pom)

What are the important files in an Angular application

- package.json : Keeps important information about the application including the dependencies (two types - normal dependencies, devDependencies) (Similar to pom.xml)
- module.ts : Holds the entire configuration of your application
- angular.json : Read after package.json. Defines the start of your project(file config??)
- Main.ts : Read after angular.json

ANGULAR WORKING

Package.json -> Angular.json -> main.ts -> app.module.ts(AppModule class)->app.component.ts(AppComponent class)->

AppModule -> main module class (The entry point) -> Needs to have definition of any new module/component we want to use in the application

AppComponent -> Every new/other component can be called only via this Component.
Example:
A and B components can be called only via AppComponent.
B can be called from A, but in that case, A HAS to be called by the AppComponent.

1.All dependencies are declared in **package.json**

2.Install the dependencies mentioned in the package.json file using '**npm install**' command in the root folder of your project. You can do it in the inbuilt terminal or cmd.

3.Start your server using the command '**npm start**' (internally runs 'ng serve'). The url to open your application is displayed if your server starts successfully.
'ng serve' internally calls 'tsc'(TypeScript Compiler) which converts all typescript code to javascript code. Reason: Browsers can only read javascript, html, css etc and not typescript
http://www.typescriptlang.org/docs/handbook/typescript-in-5-minutes.html

---

NOTE:Angular 2 onwards, **typescript(*.ts)** is used to write the code - Object based
It is similar to Java, converted into JS to be displayed on the web browser.
Before Angular 2, it was Object Oriented.
Angular 5 onwards, we use lint server.

What is One way Binding / interpolation in angular

- Either from Component to Template or vice versa
    - Component to Template
        - **Property Binding:** [property] =*value*, value)
        - **Interpolation:** {{value}}
    - Template to Component ->
        - **Event binding:** (event)=*handler* (listener)

        - The listener is a method in the component/class.
- **Two Way Binding**
    - You can access data from one into another
    - [(ngModel)] -> property
    - Always import FormsModule from @angular/forms while using ngModel
    - Internally works on 'onChange' event

Images Need to be added to "assets" folder

COMPONENTS

What is component? How can we create a component?

What is style/style url?

What are Directives? List their kinds:- **Structural ,Attributes.Components**

What is the role of the structural directives: *ngIf, *ngFor , *ngSwitch <rem syntax>

What is one way & two-way Binding?

- Interpolation. Property binding, event binding.
- Two way

What are the modules we need to import to use ngModel (FormsModule.BrowserModule)

COMPONENT LIFECYCLE - *1 QUESTION*

Create, Render, Create and render its children, checks it when its data-bound properties change, and destroys it before removing it from the DOM

Lifecycle hooks

Provide visibility into key life moments and ability to act when they occur

1. Constructor
2. **ngOnChange -**takes place when change happens
3. **ngOnInit -** set the values
4. **ngDoCheck -** check for any more changes which cannot be detected by Angular on its own. Calls the following methods only if any change is detected:
    a. **ngAfterContentInit**
    b. **ngAfterContentChecked**
    c. **ngAfterViewInit**
    d. **ngAfterViewChecked**
5. **ngOnDestroy**

Creating service class:

1. Create the service class, import Injectable from '@angular/core' and use the annotation @Injectable to declare it as service.

Benefits of service class on client side:

a. Reduce complexity of code.
b. Use the same code to access backend for all components.
c. Use the same code to perform similar/same functions common to all components.

Difference between observable and promise:
Acknowledge the request(with some token/receipt system) but respond later.

- Promise -> The proof of acknowledgement becomes invalid once the response is sent/made. Request cannot be prematurely cancelled, once acknowledged, the response WILL be sent.
- Observable -> The proof of acknowledgement can be reused any number of times. (Same token/receipt can be used to get multiple responses)
  Request can be cancelled, irrespective of the response(s) made.

Alternatives for Postman: RestAPI, SOAPUI

NOTE: No access control allow origin error : When the request received from a different server having different protocols/config

PIPES:
Transform or format data at the UI layer
Example: datepipe, currencypipe, numberpipe etc

AWS

What is Cloud?
Why do we need Cloud Services?
https://www.tutorialspoint.com/amazon_web_services/amazon_web_services_cloud_computing.htm
Types of Cloud services-
Software as a service(Saas), Platform as a service(PaaS), Infrastructure as a service(IaaS).
<There are three types of service models in cloud − IaaS, PaaS, and SaaS.>
<There are three types of clouds − Public, Private, and Hybrid cloud.>
Cloud Providers: AMAZON >>>>>Azure >>>>GCI (Market wise)

Subsidiaries	Annapurna Labs
	AWS Elemental

What is AWS? What are the services provided by AWS?
Documentation of AWS :
What is EC2? What is VPC, VPN, Security group, IAM (Identity and Access Management) and Router in Cloud

[In 2017, AWS comprised more than 90 (165 as of 2019) services. The most popular include Amazon Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (Amazon S3). Most services are not exposed directly to end users, but instead offer functionality through APIs for developers to use in their applications. Amazon Web Services' offerings are accessed over HTTP, using the REST architectural style and SOAP protocol.]

[Fun fact: Notable customers include NASA,[34] the Obama presidential campaign of 2012,[35] and Netflix, Fortnite]

What is CloudFront, Cloud Watch, Cloud Formation and Cloud Trail? Explain the difference
https://aws.amazon.com/cloudfront/
https://aws.amazon.com/cloudwatch/

What are the different Storage Services provided by aws?
What and why S3 Bucket?
What and why RDS?
What and why AWS Glacier, Storage Gateway and EBS?
What and why ElastiCache, DynamoDB?

\(OoO)/

What is CIDR? (Study calculations in IPv4 and IPv6 both) :
https://en.m.wikipedia.org/wiki/Classless_Inter-Domain_Routing _> for basics
Difference between IPv4 and IPv6 (IN DEPTH): Header format, Security
What is NAT (Network Address Translation)
:https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html
What is subnet?

DOCKER

What and why docker? :Docker is a set of platform-as-a-service products that use OS-level virtualization to deliver software in packages called containers (Wiki)

What is virtualization and Hyper-V? (Hyper-V runs each virtual machine in its own isolated space, which means you can run more than one virtual machine on the same hardware at the same time- MS Docs)

What are the Images(.iso, etc) in Docker?

How can you push or pull the images(.iso, etc) on Docker Hub?

How can we create a DockerFile?

What is Docker Container and what is Docker Engine?

How can we create a docker container?

How can we create a docker images?

What is Docker Swarm? Swami Swarmanad

MICROSERVICES

What is microservice? Describe its architecture

[Microservices - also known as the microservice architecture - is an architectural style that structures an application as a collection of services that are

- Highly maintainable and testable
- Loosely coupled
- Independently deployable
- Organized around business capabilities
- Owned by a small team

The microservice architecture enables the rapid, frequent and reliable delivery of large, complex applications. It also enables an organization to evolve its technology stack.]

What is Microservice Eureka?

How can you create a microservice with/from other microservice(s)?

What is load balancing and ribbon?