Placement Session - Web Development

Hello World!

I am Shabarish R.

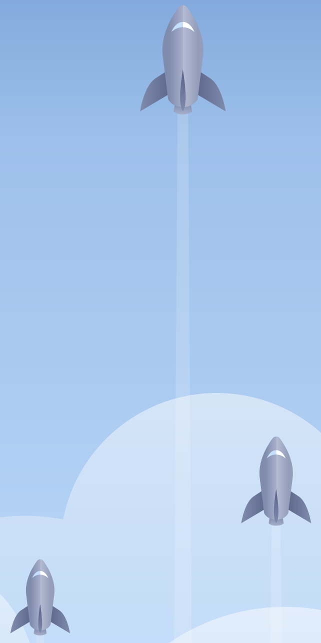
You can find me on LinkedIn.

0.

First Things First.

LinkedIn

- Make a LinkedIn Profile.
- Add all details.
- Make sure to post every once in a while.
- Add your certificates.
- Connect with seniors.
- Have a professional sounding personal email.



Resume

- Simple resume.
- Use Overleaf.
- Make it 1 Page.
- Don’t make it verbose.
- Add Extra-Curricular activities.
- Don’t write Summary.
- Do 1-2 Internships.

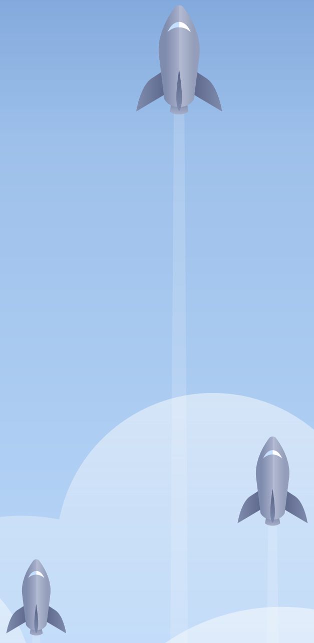
1.

Skills Required

Let’s start with the first set of skills

Skills for Web Development Placement

- Competitive Programming.
- FrontEnd Web-Dev.
- Backend Web-Dev.
- Databases.
- Hosting.
- VCS.
- Others [SoftSkills and HR round].



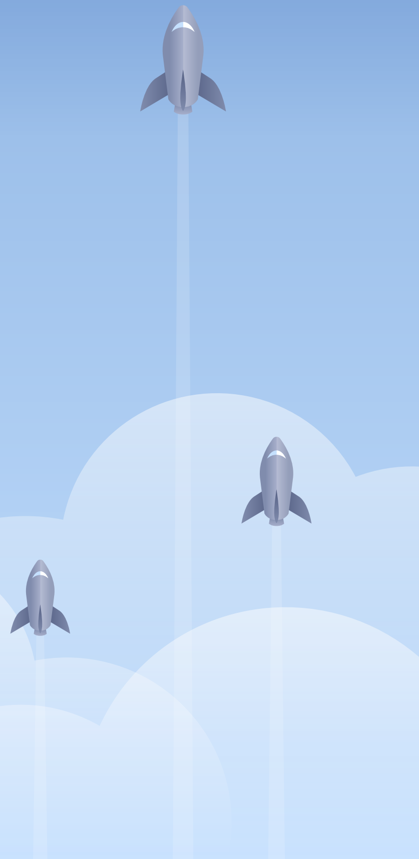
1.1.

Competitive Programming

This matters a lot !

What to in CP.

- Do all the questions in Arena.
- Practice many problems in LeetCode.
- You should be able to do LC Medium questions.
- Brush up on DS and Algorithms.
- Try to do these in language of Web Dev.



1.2.

FrontEnd

What to know in FrontEnd.

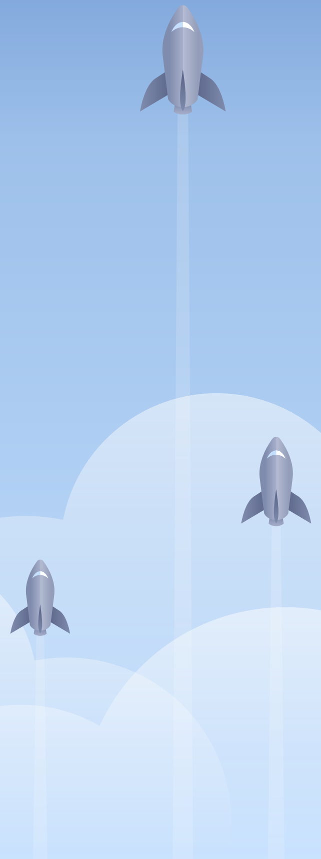
- HTML, CSS, JavaScript.
- Some SSR framework like Handlebars, EJS.
- JS Bundlers.
- ReactJS.
- TypeScript with React.

1.3.

Backend

What to know in BackEnd.

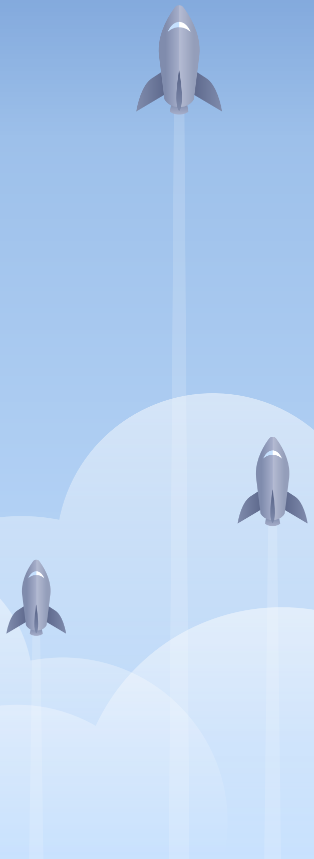
- Basics of WebServers.
- Networking Basics.
- REST, GraphQL and other Architectural styles.
- Caching strategies.
- Testing suites.
- Languages.



Choose a language

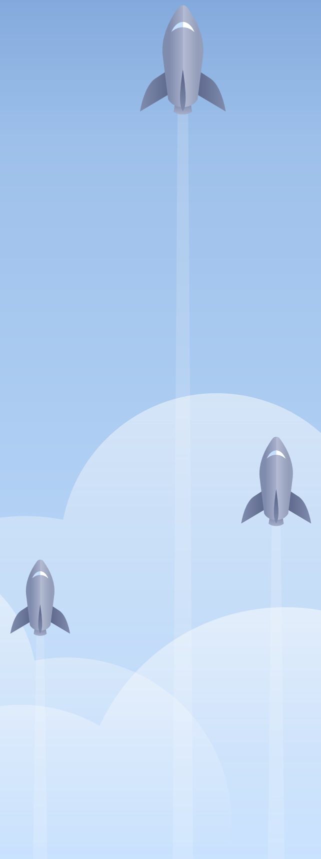
- Recommended for Beginners:
  - NodeJS and Python.

- Most used in the Industry:
  - Golang, Java, C# Dotnet, Ruby.



For NodeJS [And others]

- Asynchronous Programming.
- Event Loop.
- JS Interpreters.
- Package Management.
- Concurrency Paradigms.
- FileSystem.
- Middlewares.
- Debugging.



Basics of Backend.

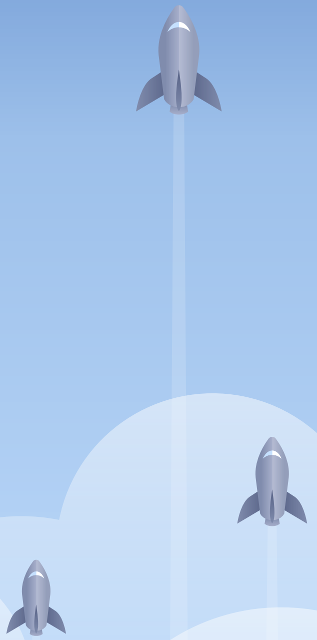
Protocols

- TCP/IP.
- HTTP, HTTPS.
- UDP.
- Sockets.
- RTP.
- SSH.

Servers

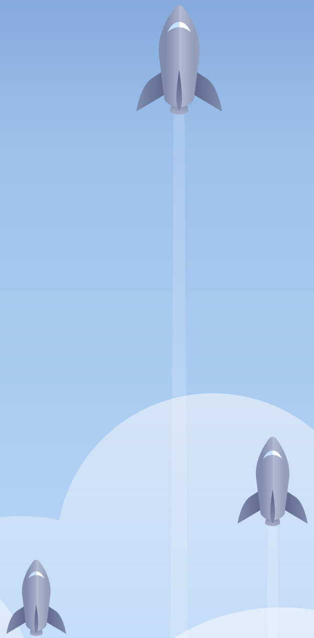
- How to build one.
- How to host one.
- Types like:
  - Forward / Reverse Proxies.
  - Cache Server.

  etc.



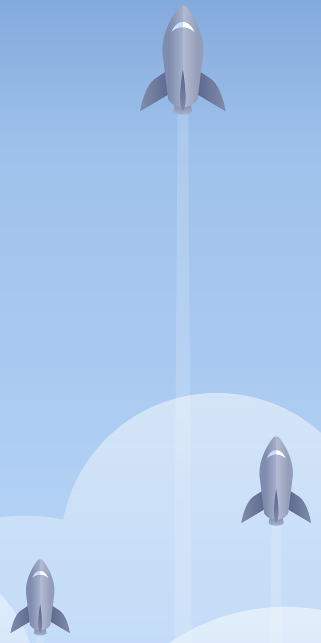
Basics of Networking.

- TCP/IP.
- DNS.
- Firewalls.
- Encryption.
- IPv4 and IPv6.
- NAT.
- QoS.
- IPSec.



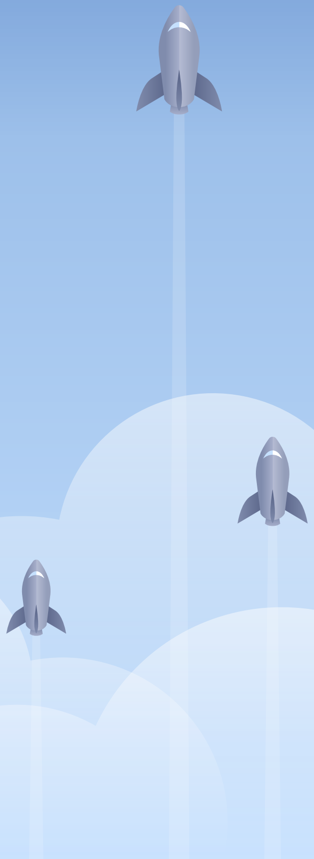
What to know in Architecture.

- SSR, SSG, CSR or other rendering styles.
- REST, GraphQL, SOAP, gRPC and API styles.
- Documentation and Extensibility.
- Caching Strategies.
- OpenAPI.
- Deployment strategies.



What to know in Testing.

- Unit Testing.
- Integration testing.
- E2E Test Automation suites.
- For NodeJS: Jest.



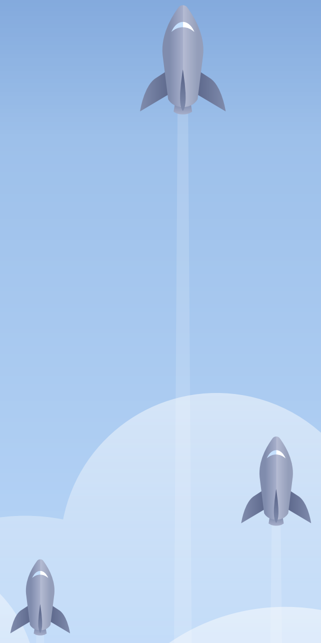
2.

Databases

Let’s start with what to know about DBs.

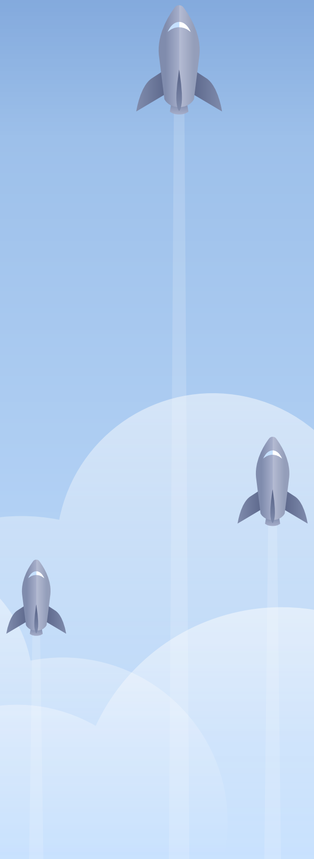
What to know in DataBases.

- Know the usecases of DBs.
- Types of DBs.
    - SQL, NoSQL, Graph, Vector etc.
- How to determine the DB for your use case.
- Object-Relational Mappers [ORMs].
- Hosted v/s local DB.



My recommendations

- SQL DB - PostGres.
- NoSQL
  - MongoDB with Mongoose.
  - Redis Cache.



3.

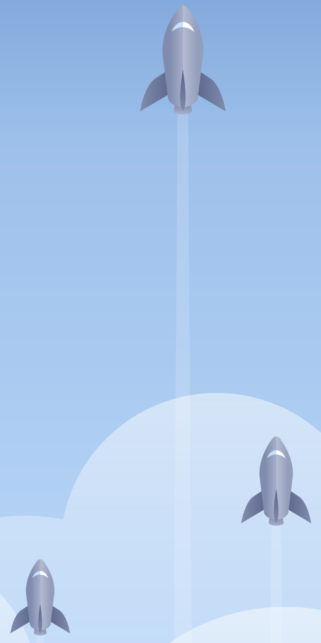
Hosting

Let’s start with what to know about Hosting.

My recommendations

- I used to use Heroku and Railway free tiers. Now they’re gone.

- Alternatives.
  - Glitch [Free].
  - AWS and Google Cloud [Paid].



4.

Project Recommendations

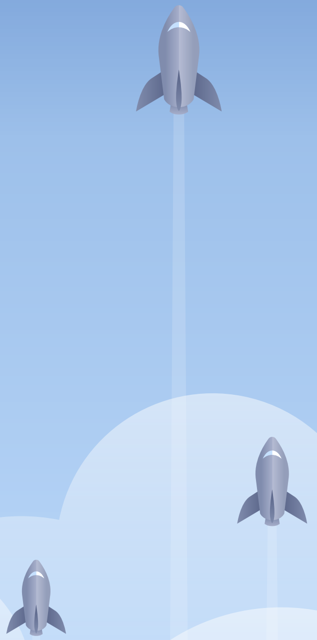
List of Project Ideas.

My recommendations

- DBMS with Dashboard.
- Real-Time chat app using Sockets.
- Fullstack application.
  - Social Media. [With Auth].
  - Exam Portal [With Forms].
- ML Application.

What all is required

- Host the application.
- Add Authentication and Authorization systems.
- Add caching.
- Build a dashboard.
- Document properly.



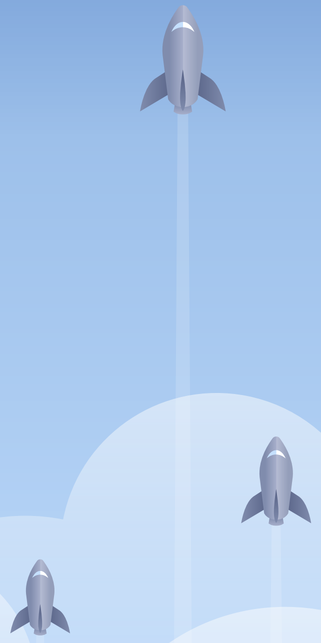
5.

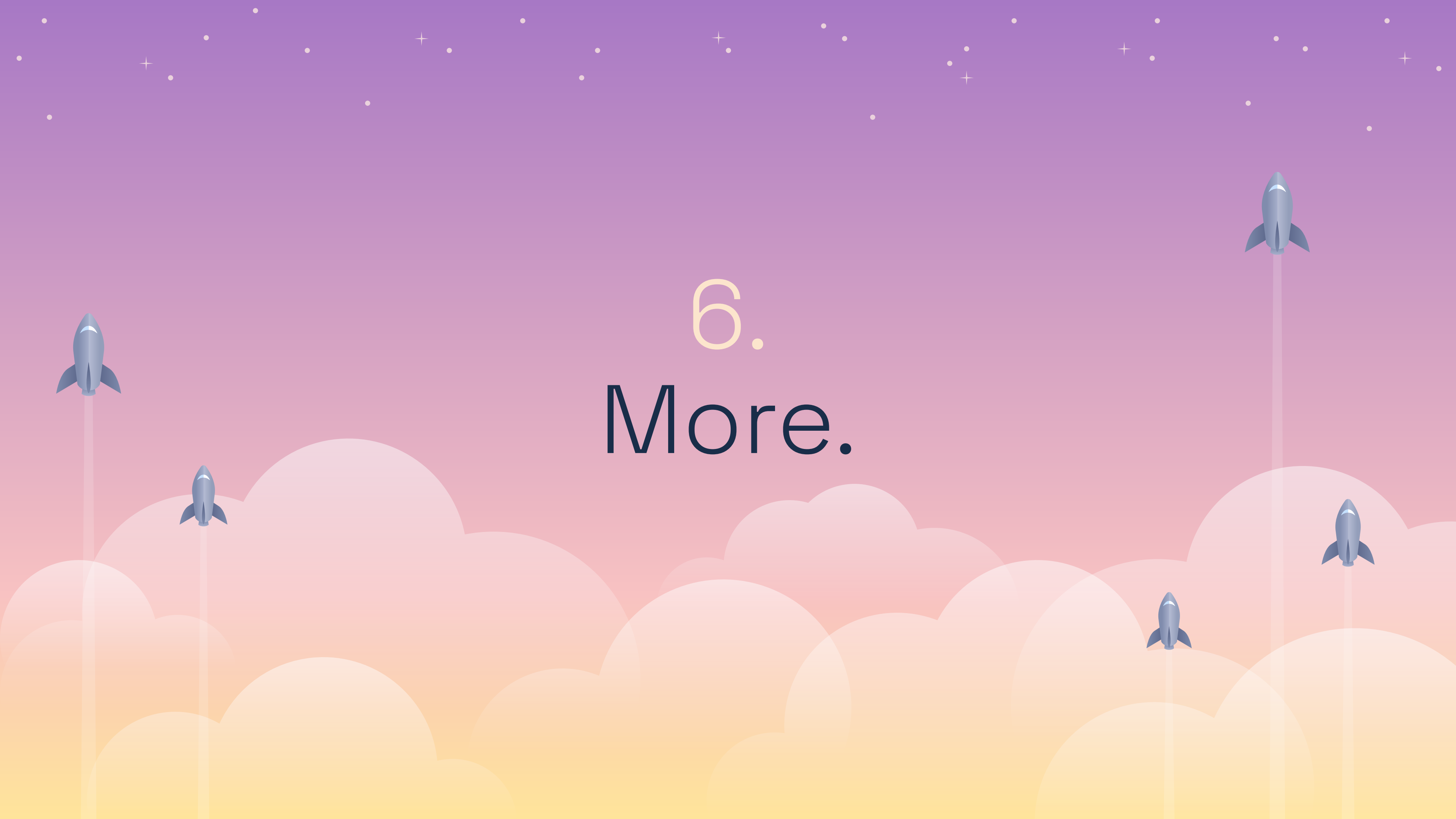
HR Round

What to say and how to say it.

What to in the HR Round.

- Prepare well and do not take it lightly.
- Do not lie.
- Stay politically neutral.
- Don’t give trite answers. Say the truth. Everyone knows you’re not gonna stay with the company forever.
- Iron out a narrative.



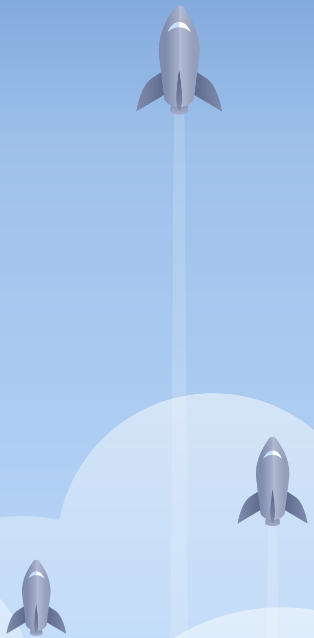


6.

More.

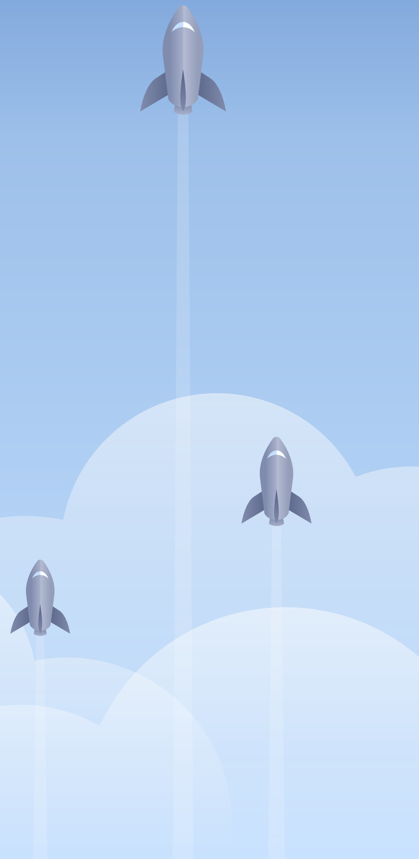
Extra

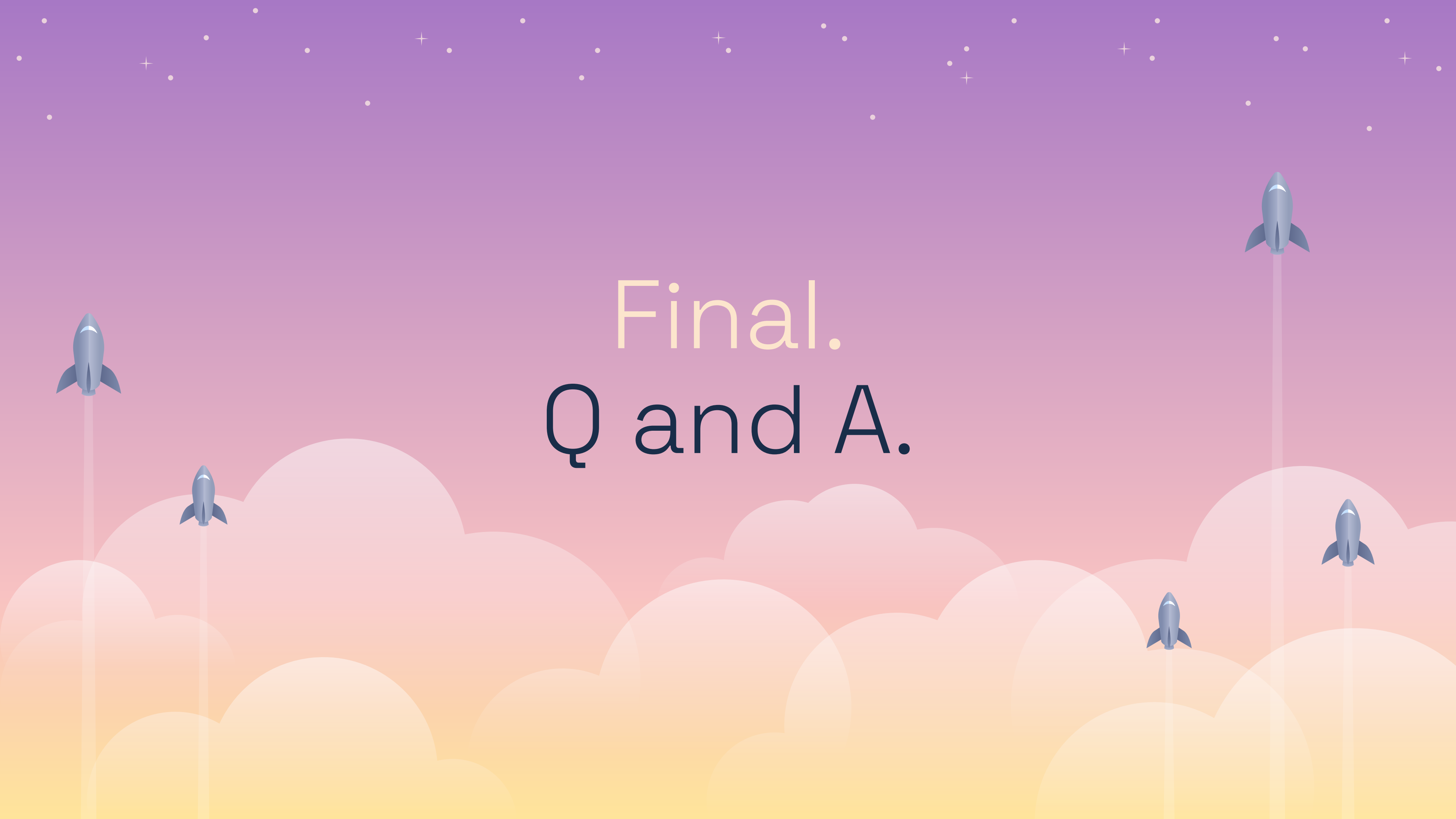
- Browsers and differences.
- WebGL, WebAssembly.
- Compression algorithms.
- Encryption and Hashing algorithms.
- Security and OWASP.
- Go through “50 most asked questions in x”.
- Linux.
- Processor Architectures.



Extra

- Coding patterns.
- Dependency Injection.
- CI/CD [Github].
- Load Balancing.
- Indexing and Searching.
- Containers and Orchestration.
- CLIs.
- Sorting algorithms.





Final.

Q and A.