Python – Backend Assignment

1. What is a Program?
   Q: Explain in your own words what a program is and how it functions.
   A: A program is a set of specific instructions written in a programming language that tells a computer what to do. It functions by being loaded into the computer's memory, where the CPU fetches the instructions one by one, decodes them , and executes them.
2. What is Programming?
   Q: What are the key steps involved in the programming process?
   A: The key steps are problem definition, algorithm and design planning, coding, testing and debugging, and maintenance.
3. Types of Programming Languages
   Q: What are the main differences between high-level and low-level programming languages?
   A: High-level languages are human-readable, portable across different computers, and easier to write, like Python. Low-level languages are machine-readable, hardware-dependent, and faster but difficult for humans to write, like Assembly.
4. World Wide Web & How Internet Works
   Q: Describe the roles of the client and server in web communication.
   A: The client, such as a web browser, initiates communication by sending a request for information. The server is a computer that stores files and processes the request, sending the required data back to the client.
5. Network Layers on Client and Server
   Q: Explain the function of the TCP/IP model and its layers.
   A: The TCP/IP model is a set of protocols used to connect devices on the internet. Its layers are the Application layer for user interfaces, the Transport layer for reliability, the Internet layer for routing, and the Network Access layer for physical connections.
6. Client and Servers
   Q: Explain Client Server Communication.
   A: Client Server Communication is a structure where tasks are divided between servers, which provide resources, and clients, which request them. It follows a Request-Response cycle.
7. Types of Internet Connections
   Q: How does broadband differ from fiber-optic internet?
   A: Broadband typically refers to cable internet using copper wires and electrical signals. Fiber-optic uses glass cables to transmit data as pulses of light, making it significantly faster and less prone to interference.
8. Protocols
   Q: What are the differences between HTTP and HTTPS protocols?
   A: HTTP transmits data in plain text, meaning it can be read by hackers if intercepted. HTTPS uses encryption to scramble the data, making it secure for sensitive activities like banking.

9. Application Security
   Q: What is the role of encryption in securing applications?
   A: Encryption converts readable data into an encoded format that can only be read by someone with a decryption key. This protects sensitive information from being stolen during transmission.
10. Software Applications and Its Types
   Q: What is the difference between system software and application software?
   A: System software, like Windows or macOS, runs the computer hardware. Application software, like Word or Chrome, is designed for the user to perform specific tasks.
11. Software Architecture
   Q: What is the significance of modularity in software architecture?
   A: Modularity breaks a system into independent parts called modules. This makes the software easier to understand, allows simultaneous development, and makes fixing bugs easier without affecting the whole system.
12. Layers in Software Architecture
   Q: Why are layers important in software architecture?
   A: Layers separate different parts of the code, such as the user interface and the database. This ensures that changes in one layer do not force the programmer to rewrite code in other layers.
13. Software Environments
   Q: Explain the importance of a development environment in software production.
   A: A development environment allows programmers to write and test code safely without affecting the real users. It helps catch bugs before the software goes live.
14. Source Code
   Q: What is the difference between source code and machine code?
   A: Source code is the code written by humans in programming languages. Machine code is binary consisting of zeros and ones that the computer processor executes directly.
15. Github and Introductions
   Q: Why is version control important in software development?
   A: Version control tracks every change made to the code. It allows developers to undo mistakes, revert to previous versions, and manage conflicts when multiple people work on the same file.
16. Student Account in Github
   Q: What are the benefits of using Github for students?
   A: Github provides students with free access to professional tools, helps them build a portfolio to show employers, and teaches them how to collaborate on coding projects.
17. Types of Software
   Q: What are the differences between open-source and proprietary software?
   A: Open-source software code is free for anyone to view and modify. Proprietary software is owned by a company, the code is secret, and users usually pay for a license.
18. GIT and GITHUB Training
   Q: How does GIT improve collaboration in a software development team?
   A: Git allows developers to use branches to work on different features at the same time without overwriting each other's work. The changes are then merged together.

19. Application Software
    Q: What is the role of application software in businesses?
    A: Application software increases productivity by automating tasks, managing customer data, analyzing business performance, and helping employees communicate.
20. Software Development Process
    Q: What are the main stages of the software development process?
    A: The main stages are planning, requirements analysis, design, implementation or coding, testing, deployment, and maintenance.
21. Software Requirement
    Q: Why is the requirement analysis phase critical in software development?
    A: This phase defines exactly what needs to be built. If requirements are wrong, the final software will not solve the user's problem, leading to failure.
22. Software Analysis
    Q: What is the role of software analysis in the development process?
    A: Software analysis ensures that the technical solution being planned actually aligns with the business needs and goals.
23. System Design
    Q: What are the key elements of system design?
    A: The key elements are system architecture, data design, interface design, and component design.
24. Software Testing
    Q: Why is software testing important?
    A: Testing ensures the software works correctly and is free of bugs. It improves quality and saves money by fixing issues before the software is released.
25. Maintenance
    Q: What types of software maintenance are there?
    A: The types are corrective maintenance to fix bugs, adaptive maintenance for new environments, perfective maintenance to add features, and preventive maintenance to clean up code.
26. Development
    Q: What are the key differences between web and desktop applications?
    A: Web applications run in a browser and require the internet. Desktop applications are installed on a computer and run locally, often without needing the internet.
27. Web Application
    Q: What are the advantages of using web applications over desktop applications?
    A: Web applications do not require installation, can be accessed from any device, and are updated instantly for all users without needing downloads.
28. Designing
    Q: What role does UI/UX design play in application development?
    A: UI design focuses on how the application looks, while UX design focuses on how it feels and functions. Good design ensures the user is satisfied and can use the app easily.
29. Mobile Application
    Q: What are the differences between native and hybrid mobile apps?
    A: Native apps are built for a specific operating system like iOS and offer high performance. Hybrid apps are web apps wrapped in a container that run on multiple platforms but may be slower.

30. DFD (Data Flow Diagram)
    Q: What is the significance of DFDs in system analysis?
    A: Data Flow Diagrams visually map how data enters, flows through, and leaves a system, helping developers understand the logic before coding.
31. Desktop Application
    Q: What are the pros and cons of desktop applications compared to web applications?
    A: Desktop apps are faster and work offline but are harder to update and require specific versions for different operating systems.
32. Flow Chart
    Q: How do flowcharts help in programming and system design?
    A: Flowcharts provide a visual map of the logic. They help programmers see the path of the code and find logical errors before writing the actual syntax.