**Q1. Name the programming languages that are used to develop backend applications!**

* **Ans:** Back-end languages include Python, Java, C#, PHP, JavaScript, Golang, C++, SQL, Kotlin, Ruby, etc. A server-side developer is an expert that uses one of these programming languages to develop the back end of a web or mobile app.
* **Reference:** <https://www.google.com/search?q=Name+the+programming+languages+that+are+used+to+develop+backend+applications!&oq=Name+the+programming+languages+that+are+used+to+develop+backend+applications!&gs_lcrp=EgZjaHJvbWUyBggAEEUYOdIBBzI1NWowajeoAgCwAgA&sourceid=chrome&ie=UTF-8>
* **Q2. Name and mention various technologies that need to be studied to become a backend engineer!**
* **Ans:** To be a successful backend developer, you need to have a good understanding of various technologies and programming languages. Here are some key technologies and concepts that are commonly used in backend development:

1. **Programming Languages**:  
   - **Java**: Widely used in enterprise development.  
   - **Python**: Known for its simplicity and readability.  
   - **JavaScript (Node.js)**: Allows you to use JavaScript on the server-side.  
   - **Ruby (Ruby on Rails)**: Known for its simplicity and elegance.  
   - **C#**: Commonly used in Microsoft environments.
2. **Databases**:  
   - **SQL (MySQL, PostgreSQL, SQLite)**: Traditional relational databases.  
   - **NoSQL (MongoDB, Redis)**: Non-relational databases for handling large volumes of data.
3. **Web Frameworks**:  
   - **Spring (Java)**: Popular Java framework for building enterprise applications.  
   - **Django (Python)**: High-level Python web framework.  
   - **Express (Node.js)**: Minimal and flexible Node.js web application framework.
4. **APIs (Application Programming Interfaces)**:  
   - Understanding how to design and consume APIs is crucial for backend development.
5. **Version Control Systems**:  
   - **Git**: Essential for tracking changes in code and collaborating with other developers.
6. **Web Servers**:  
   - **Apache, Nginx**: Common web servers used to host web applications.
7. **Authentication and Authorization**:  
   - Understanding concepts like OAuth, JWT, and role-based access control.
8. **Testing**:  
   - **Unit Testing**: Writing tests to verify the functionality of individual components.  
   - **Integration Testing**: Testing interactions between different components.
9. **DevOps Tools**:  
   - **Docker**: Containerization tool for packaging applications.  
   - **Continuous Integration/Continuous Deployment (CI/CD)**: Tools like Jenkins, Travis CI, or GitLab CI for automating the deployment process.
10. **Security**:
    * Understanding common security vulnerabilities and best practices for securing applications.
11. **Cloud Services**:
    * Familiarity with cloud platforms like AWS, Azure, or Google Cloud for deploying and scaling applications.
12. **Microservices Architecture**:
    * Understanding how to design and implement microservices for building scalable and maintainable applications.

**Reference:** <https://www.quora.com/What-kind-of-technology-I-have-to-know-to-be-a-backend-developer>