Q.2. What is difference between High Level Language Vs Low Level Language?

→

High-Level Languages	**Low-Level Languages**
1. Designed to be easy for humans to read and write. 2. Abstract away most of the hardware details, allowing focus on problem-solving and logic. 3. Examples include Python, Java, and C++. 4. Use natural language elements and clear syntax. 5. Easier to learn, understand, and maintain.	1. Closer to machine code and hardware. 2. Provide minimal abstraction, offering direct control over hardware. 3. Examples include Assembly language and machine code. 4. Use mnemonic codes and hexadecimal or binary numbers. 5. More challenging to read and write. 6. Ideal for system programming and performance-critical applications. 7. Provide low abstraction, with

6. Suitable for complex applications and large codebases. 7. Offer high abstraction from hardware.	direct access to memory and CPU instructions.