Java Project flow

① Spring boot appln → initializes Tomcat server
→ scans classpath for components.

② Component Scanning: @RestController
@Service
@Repository
@Entity

③ Request Handling
When a client sends an HTTP request to appln, the request first intercepted by the embeded Tomcat Server.
Then Tomcat routes the request to the appropriate Controller based on the request URL & HTTP method.

④ Controller Mapping @RestController
Controller classes handles the Incoming HTTP requests
Methods within controller classes are annotated with @GetMapping @Post @Put @DeleteMapping to specify the URL patter & & HTTP method they handle.

⑤ Request Parsing When request is recieved by the controller method, Spring boot automatically binds request parameters, headers & body to method parameters like @PathVariable, @RequestBody, @RequestParam etc.
Controller method invokes Service layer method to perform Business logic.

⑥ Service Layer @Service
encapsulates the appln Business logic
Service layer interacts with the Repositories to perform CRUD operations on Data.

⑦ Data Access Layer @Repository
DAL interact with the db.
Spring Data JPA → JPA Repository

Model Layer POJO → to represent data entities.
@Entity.

⑧ Response Generation

Once the necessary operations are completed, the service layer returns the result to the controller, which generates an appropriate HTTP response.
This response includes relevant data and an appropriate HTTP status code.

⑨ Response Sending

The generated response is sent back to the client by the embeded Tomcat Server.