

## **Pentonix Assesement 2 Explanation**

Step 1: Similar to the last code, these are the package and import statements.

Step 2: The static maps for storing department and personnel information are defined in the main class of the program.

Step 3: The DepartmentHandler object is connected to the /api context via the main method, which also establishes an HTTP server listening on port 9000 and loads the Department and Employee databases.

Step 4: The structure of an Employee is described by this class. It has attributes for the name, department number (dno), employee ID, and compensation. In order to access these attributes, it also includes getter methods.

Division Class:

Step 5: This class describes a department's organizational structure, including the department name (dname) and department number (dno). It offers getter techniques for reaching these properties.

Step 6: In order to process incoming HTTP requests, this inner static class implements the HttpHandler interface. In order to process the request, the handle method takes the ENO from the query input and looks for the information for the matching employee. If a match is made, it generates a Department instance and replies with the employee's information and the department name.

During operation, a new HTTP server on port 9000 is created and configured to monitor requests made to the /api endpoint. A client's request with an ENO query parameter causes the server to search up the employee's information and department name. If the employee is located, the server replies with the employee's information and the name of the department. The proper error replies are issued if the employee cannot be located or the request is insufficient.