

PRACTICE 3 OOPS+CORE JAVA LOGIC QUESTION+SQL +SPRINGBOOT -REST APIS

OOPs+Logic

Concept : Encapsulation

Encapsulation means wrapping data and methods together and hiding data using private access.

```
class Student{

    //encapsulate this data as using private access modifier

    private int id;

    private String name;

    //getter and setter

    public void setId(int id){

        this.id=id;

    }

    public int getId(){

        return id;

    }

    public void setName(String name){

        this.name=name;

    }

}
```

```
public String getName(){  
    return name;  
}  
  
public class Test{  
    public static void main(String[] args){  
        //object  
        Student s = new Student();  
        s.setId(1);  
        s.setName("Geetanjali");  
  
        System.out.println(s.getId);  
        System.out.println(s.getName);  
  
    }  
}  
}
```

Java core logic

```
// find the largest number in array
```

```
int[] arr = {10,45,2,89,30};  
int max = arr[0];  
  
for(int i =1; i<arr.length; i++){  
    if(arr[i]>max){  
        max = arr[i];  
    }  
}
```

```
}
```

```
}
```

```
System.out.println("Largest" +max);
```

SQL question

1. create table

```
CREATE TABLE employee(  
id INT PRIMARY KEY,  
name VARCHAR(50),  
salary INT,  
deparment VARCHAR(30)  
);
```

2. Insert Data

```
INSERT INTO employee VALUES (1, 'John', 40000, 'IT');  
INSERT INTO employee VALUES (2, 'Aman', 30000, 'HR');
```

// Important interview Query

```
Select * from employee;
```

Select name, salary from employee;

select * from employee where salary > 35000;

select * from employee Orderby salary DESC;

//diff between where and having

where filter rows before grouping

having filters after groupby

//Springboot Rest APIS

1. Simple REST Controller

@RestController

@RequestMapping("/api")

public class TestController{

@GetMapping("/hello")

public String hello(){

return "Hello from SPRINGBOOT";

}

}

2. REST API for Student

```
@RestController
@RequestMapping("/student")

public class StudentController{

    @GetMapping("/{name}")
    public String getStudent(@PathVariable String name){
        return "Student name is "+name;
    }
}
```

3. URL

<http://localhost:8080/student/John>

//Mini practice

Java: Create a class Employee with private fields

```
class Employee{
    private int id;
    private String employeeName;
```

```
public void setId(int id ){  
    this.id=id;  
}  
  
public int getId(){  
    return id;  
}  
  
public void setName(String name){  
    this.name=name;  
}  
  
public int getName(){  
    return name;  
}  
  
public class EmployeeTest{  
    public static void main(String[] args){  
        //create a object  
        Employee emp = new Employee();  
        emp.setId(1);  
        emp.setName("Riya");  
        System.out.println(emp.getId);  
        System.out.println(emp.getName);  
    }  
}
```

//output is 1 and riya

SQL: Write query to get highest salary

```
select * from employee where salary >50000;
```

Spring Boot: Create /api/welcome GET API

```
@RestController
```

```
@RequestMapping("/welcome")
```

```
public class TestController{
```

```
    @GetMapping("/Welcome")
```

```
    public String hello(){
```

```
        return "Hello from spring boot";
```

```
    }
```

```
}
```