

## Spring Boot API without JDBC

### 1. Get String data

```
@GetMapping("/firstapi")
public String StringAPI()
{
    return "Hello Spring Boot Application!";
}
```

### 2. Get integer data

```
@GetMapping("/intvalue")
public int IntAPI2()
{
    return 10;
}
```

### 3. Get Integer Array

```
@GetMapping("/array")
public int[] ArrayAPI3()
{
    int array[]={ 10,20,30,40,50};

    return array;
}
```

### 4. Get String ArrayList

```
@GetMapping("/arraylist")
public ArrayList<String> ArrayListAPI4()
{
    ArrayList<String> arraylist=new ArrayList<>();
    arraylist.add("abc");
    arraylist.add("pqr");
}
```

```
        arraylist.add("str");  
  
        return arraylist;  
    }  
}
```

## 5. Get Employee type data

```
@GetMapping("/employeeapi")  
public Employee1 EmployeeAPI8()  
{  
    Employee1 emp=new Employee1();  
    emp.setAge(30);  
    emp.setName("Abc");  
  
    return emp;  
}
```

## 6. Get Employee type arraylist

```
@GetMapping("/arraylistemployeeapi")  
public ArrayList<Employee1> EmployeeAPI9()  
{  
    Employee1 emp=new Employee1();  
    emp.setAge(30);  
    emp.setName("Abc");  
  
    Employee1 emp1=new Employee1();  
    emp1.setAge(25);  
    emp1.setName("Pqr");  
  
    ArrayList<Employee1> emparraylist=new ArrayList<>();  
    emparraylist.add(emp);  
    emparraylist.add(emp1);  
  
    return emparraylist;  
}
```

## 7. Post String value

@PostMapping("/poststring")

```
public String getString(@RequestBody String post)
{
    System.out.println(post);
    return "Value added successfully!";
} //give input e.g. hello world
```

## 8. Post Integer value

@RequestMapping(value="/postint", method=RequestMethod.**POST**)

```
public String getInteger(@RequestBody String post)
{
    System.out.println(post);
    return "Value added successfully!";
} //give input e.g. hello world
```

## 9. Put String value

@PutMapping("/putpost")

```
public String putString(@RequestBody String post)
{
    String value;
    value="Hello world";
    value=post;
    System.out.println("Updated value:-"+value);
    return "Value updated!";
}
```

## 10. Put Integer Value

@RequestMapping(value="/putpostint",  
method=RequestMethod.**PUT**)

```
public String putString(@RequestBody int post)
{
    int a;
    a=10;
    a=post;
```

```
System.out.println("Updated value:-"+a);  
return "Value updated!";  
}
```

## 11. Get Hash set

```
@GetMapping("/hashsetapi")  
public HashSet<Integer> HashSetAPI10()  
{  
    HashSet<Integer> set=new HashSet<>();  
    set.add(10);  
    set.add(20);  
    set.add(5);  
    return set;  
}
```

## 12. Get Treeseet

```
@GetMapping("/treesetapi")  
public TreeSet<Integer> TreeSetAPI11()  
{  
    TreeSet<Integer> set=new TreeSet<>();  
    set.add(10);  
    set.add(25);  
    set.add(5);  
    set.add(60);  
    set.add(40);  
  
    return set;  
}
```

## 12. Check Employee Exist or not.

```
@RequestMapping("/getemployeeapi/{id}")
public String getEmployeebyid(@PathVariable String id)
{
    int id1=Integer.parseInt(id);
    Employee1 emp=new Employee1();
    emp.setId(1);
    emp.setAge(30);
    emp.setName("Abc");

    List<Employee1> listemployee=new ArrayList<>();
    listemployee.add(emp);
    boolean flag=false;
    for(Employee1 employee:listemployee)
    {
        if(employee.getId()==id1)
        {
            flag=true;
        }
    }
    if(flag==true)
    {
        return "Employee found";
    }
    else
    {
        return "Employee doesn't exist";
    }
}
```