# Interview question with answer

## 1. List down all the annotation in spring.

Answer: These are the basic annotation presents in spring

- @Component
- @Service
- @Controller
- @Repository
- @Require
- @Autowired
- @Qualifier
- @Configuration
- @Bean
- @ComponentScan
- @Import
- @Dependson
- @Lazy
- @RequestParm
- @PathVariable
- @ModelAttribute
- @RequestMapping
- @Transactional
- @AdviceController
- @ExceptionHandler

## 2. List out all annotation of rest used in your projects?

Answer: In my project I am using these below annotations

1. @ApplicationPath
2. @Path
3. @Consumes
4. @Produces
5. @POST
6. @GET
7. @PUT

8. @DELETE
9. @FormParm
10. @PathParm etc…

### 3. Use of cascade and inverse attributes in Hibernate?

Answer:

Cascade attribute normally used to save child along with parent, simply in One to many relationship one parent can contain List of child so to save list of child along with parent we have to mention cascade attribute with value 'all'

Inverse attribute is used to change the relationship of owner, and default value of inverse='false' if we assign the value is inverse='true' then in case of update and save operation it will not reflect in child it always showing the changes of parent

### 4. Difference between Level 1 and level 2 cache in Hibernate?

| **Level-1 cache** | **Level-2 cache** |
|---|---|
| 1. Level-1 cache is session scope. | 1.Level-2 cache is Session Factory scope |
| 2. By default enable. | 2.We need to enable it by using some configuration |
| 3.level 1 cache data will not available in entire application cause its scope is session scope and one application can contain multiple session | 3. level 2 cache data will available in entire application cause its scope is session factory scope |
| 4.It implements SRAM (static random access memory) | 4.It implements DRAM(Dynamic random access memory) |
| 5.It have less memory | 5.It have more memory than Level-1 |
| 6.It is Faster than Level-2 | 6.Process is little bit slow than Level-1 |

### 5. Difference between ArrayList and LinkedList ?

| ArrayList | LinkedList |
|---|---|
| 1.For retrieval operation ArrayList is preferable | 1.For Middle insertion or middle modification LinkedList is preferable |
| 2.Memory shifting or Memory management is complicated | 2.Memory shifting or Memory management is not required |
| 3. ArrayList internally uses **dynamic array** to store the elements. | 3. LinkedList internally uses **doubly linked list** to store the elements. |

## 6. Internal flow of HashMap ?

Normally HashMap follows below 3 steps

1. hashCode() to find bucket

2. == operator to reference comparison

3. equals () to content comparison

When we are save object in map means map.put(Object,Object) then first hashMap internally use hashing technique to find the bucket so it find the hashCode of object then % by initial capacity then bucket number will be get and object will placed on that specific bucket no Ex: (key.hashCode()%11)

After placing object if in same bucket 2 objects are placed then there may be a chance key collision .to avoid this situation it internally uses == operator so here it will check whether these 2 objects have same hashCode or not? If hashCode is different then it will place the object in map else if both have same hashcode then there may be chance hashing collision.

If hashing collision occurs then to overcome this issue it internally use  equals method to content comparison  it check obj1.equals(obj2) if content same then replace the value else add that object into map.

### 7. Difference between hashMap and hashTable ?

| HashMap | HashTable |
|---|---|
| 1.HashMap is non-synchronized and non-thread safe | 1.HashTable is synchronized and thread safe |
| 2.HashMap allow only one null as key and multiple null we can take as value | 2.Null insertion is not there in HashTable |
| 3.we can use Iterator in HashMap | 3.Iterator is not support ,here we have to use Enumeration |
| 4.HashMap is faster and consumes less memory than hashTable | 4.HashTable is not faster and it consumes more memory than HashMap |
| 5.HashMap is introduced 1.2 version | 5.HashTable is introduced in 1.2 so it is legacy |

### 8. Difference between Array and ArrayList ?

| Array | ArrayList |
|---|---|
| 1. Array is not Grow able in nature. | 1.ArrayList is grow able in nature |
| 2.It allow only homogeneous object | 2.It allow both homogeneous and heterogeneous object |
| 3.Inbuilt methods are not there in Array | 3.Inbuilt methods are there provided by utile class |
| 4.Performance wise Array is best | 4.Performance wise ArrayList is preferable |

| | |
|---|---|
| 5.Memory wise Array is not good | 5.Memory wise ArrayList is not good in comparison Array |

### 9. Difference between ArrayList and vector ?

| **ArrayList** | ***Vector*** |
|---|---|
| 1.ArrayList is non-synchronized | 1.Vector is synchronized |
| 2.It is Non-Thread safe | 2.it is thread safe |
| 3.it Introduced in jdk 1.2 | 3.Vector is legacy |
| 4.ArrayList increments 50% of its current array size If number of element exceed its initial capacity | 4.It increment double of its current initial capacity |
| 5.Support Iterator | 5.Support Enumeration |

### 10. in your project where you used ConcurrentHashMap?

Answer:

As per my project business requirement I didn't use ConcurrentHashMap that's why I don't have hands on experience on it but I know where we have to use.

Suppose I have one requirement where multiple thread want to access/modify under laying DS of map means one thread try to insert element in key and value pair and another thread try to modify some element at same times then in this situation better to prefer ConcurrentHashMap instead of HashMap cause ConcurrentHashMap apply lock on specific entry not in complete map object.

**NOTE: Normally to avoid ConcurrentModificationException ConcurrentHashMap introduced in JDK 1.5**

### 13. Difference between Callable and Future interface?

Answer:

Both are introduced in java.util.concurrent package in 1.5 v and we are using these 2 interfaces in Executor framework.

Callable interface is used for create thread and the main advantages of using Callable is after completion of thread execution it returns object cause the return type of call method is Object

```
public Object call() throws Exception {
        return null;
  }
```

Future I is a general concurrency abstraction, also known as a promise, which promise to return a result in future. When we are calling service.submit(Pass Thread class Object); then it instantly computation the result and return back to the future object based on generics and we can get that results by calling future.get() method

```
ExecutorService service = Executors.newFixedThreadPool(2);
 Future<Object> f = service.submit(Thread class Object);
 Object result = f.get();
```

## 14. Class loaders?

Answer: Class Loaders helps to load the class in JVM normally there are 3 types of class loader available i.e.  1. Bootstrap Class Loader  2.Extension Class Loader  3.Application Class Loader

These class loader follows 3 principle to load the class see the below description

1. Principle of delegation

2. Principle of visibility

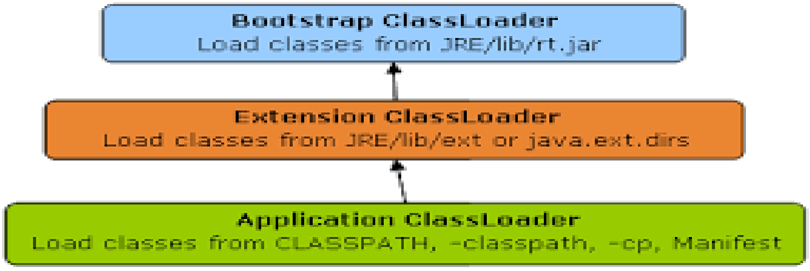
3. Principle of uniqueness

**Principle of delegation:**

Normally at the time of class loading always control comes to child class loader i.e. Application Class Loader as per this principle it forward the request to its parent class loader i.e. extension class loader then again extension class loader forward it to its parent class loader Bootstrap Class Loader now Bootstrap Class Loader search from his directory to find the .class file if find then load else next hierarchy class loader i.e. extension class loader search in his directory if find then it will load by this class loader else again it goes to next hierarchy class loader i.e. Application class loader if found the it will be load else we will get ClassNotFoundException.

**Principle of visibility:**

This principle maintains the parent and child relationship as we know a child can know the status of parent class but a parent class can’t know the status of its child same here see the below hierarchy.


Bootstrap ClassLoader
Load classes from JRE/lib/rt.jar
Extension ClassLoader
Load classes from JRE/lib/ext or java.ext.dirs
Application ClassLoader
Load classes from CLASSPATH, -classpath, -cp, Manifest

**Principle of uniqueness:**

Principle of uniqueness means if already class is loaded by one class loader it can’t be again reload by other otherwise ambiguity will be occurs let’s assume suppose I have one class and I placed this class in JRE/lib/ext folder so then this class will be loaded by whom ?

As per above two principle first request will be delegate to its corresponding parent so after 2 search finally control goes to Extension class loader where my .class file is available so control can’t goes to application class loader that’s why class will be loaded by Extension class loader. Here already my .class loaded by Ext loader so it never again load by subclass loader Application Class loader as per Principle of uniqueness

## 15. How can we take list into map?

Example: `public Map<List<String>, List<Double>> getData() {`

```
             Map<List<String>, List<Double>> dataMap = new HashMap<>();

             List<String> l1Key = new ArrayList<>();
             l1Key.add("Basant");
             l1Key.add("Babul");

             List<String> l2Key = new ArrayList<>();
             l2Key.add("Manoj");
             l2Key.add("Amit");
             l2Key.add("Saroj");

             List<Double> l1Value = new ArrayList<>();
             l1Value.add(50.000);
```

```
		l1Value.add(40.000);

		List<Double> l2Value = new ArrayList<>();
		l2Value.add(90.000);
		l2Value.add(60.000);
		l2Value.add(30.000);

		dataMap.put(l1Key, l1Value);
		dataMap.put(l2Key, l2Value);
		return dataMap;

	}
```

## 16. How can we take Map into List?

Example:

```
	public List<Map<String, Integer>> setData() {
		List<Map<String, Integer>> listMap = new ArrayList<>();
		Map<String, Integer> map = new HashMap<>();
		map.put("A", 34);
		map.put("D", 90);
		map.put("B", 12);
		map.put("C", 91);
		listMap.add(map);

		return listMap;
	}
```

## 17. When you will get ClassNotFoundException and NoClassDefFoundError ?

Answer:

When our class loaded by class loader but it will not found in class path then ClassNotFoundException will be raised ex: FileNotFoundException, when you are trying to perform any file operation but your file is not present in project directory then u will get FileNotFoundException

When class is visible only at compilation time but not available in Jvm/Runtime ,then in that situation we will get NoClassDefFoundError.

## 18. .how you implement exception handling in your project?

Answer:

Normally we are developing Spring based application so in Spring to handle exception multiple predefined class is there .so simply in my project we are throwing custom exception from service layer and when my controller call service it will catch that exception by using Spring-

Aop , We have to create a class which should be annoted as @ControllerAdvice and we have to take one method whose return type is Model And View and method should be annoted as @ExceptionHandler so in this method we have to write the logic for map the exception. And return the same view which is return by controller class at the time of exception raise. And we have to return some user understandable message by view page.

**Note: Both return logical view should be same**

## 19. Where you implement multi-threading in your project?

Answer:

As a 3 year experience developer I didn't get a chance to work on Multithreading environment .these part is developed by our senior team members. But I have aware on multithreading and I know how to work.

## 20. What are all the design patterns you observed in spring?

Answer:

As I know in spring internally used 4 to 5 annotations to provide better flexibility to developer

1. Strategy Design Pattern

2. Factory Design Pattern

3. Template Design Pattern

4. Adapter Design Pattern

5. Singleton Design Pattern

6. Facade Design Pattern

## 21. Which design patters you used in your project?

Answer:

In my project I used Modularization design Pattern, Facade Design Pattern, Service locator Design pattern.

## 22. What are all the critical situations you come across in your project?

## 23. Why wait () placed in object class? Why not it is placed in Thread class?

Answer:

Very simple reason behind it actually wait () method is used to wait the thread of execution and it used for inter thread communication to avoid data inconsistency, so first reason is wait () and notify () method are not specific for single object it can be apply, second reason is if it will present in thread class then one thread have to know the status of another thread

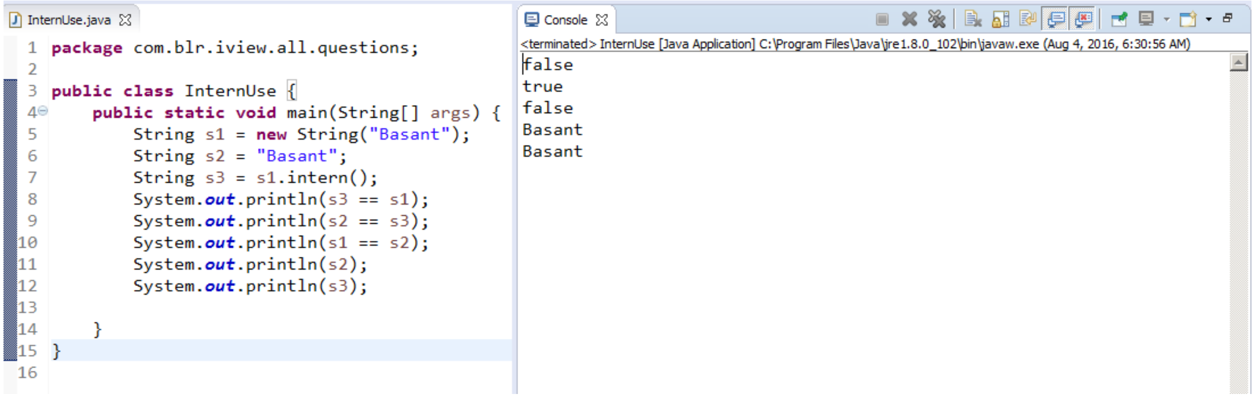
**Example**: Suppose I have 4 thread th1 ,th2 , th3, th4 and th4  so in inter thread communication lock will be applied on Object by thread so which thread apply lock that only known to object .simply locking mechanism no way related to thread its related to Object.

## 24. What is use of intern () in string?

Answer:

Intern () method is used to avoid string duplication. And if you want to get the reference of string instance from SCPA with the help of heap reference we have to use intern () method.

**Example**:



```
package com.blr.iview.all.questions;

public class InternUse {
    public static void main(String[] args) {
        String s1 = new String("Basant");
        String s2 = "Basant";
        String s3 = s1.intern();
        System.out.println(s3 == s1);
        System.out.println(s2 == s3);
        System.out.println(s1 == s2);
        System.out.println(s2);
        System.out.println(s3);

    }
}

```

```
<terminated> InternUse [Java Application] C:\Program Files\Java\jre1.8.0_102\bin\javaw.exe (Aug 4, 2016, 6:30:56 AM)
false
true
false
Basant
Basant
```

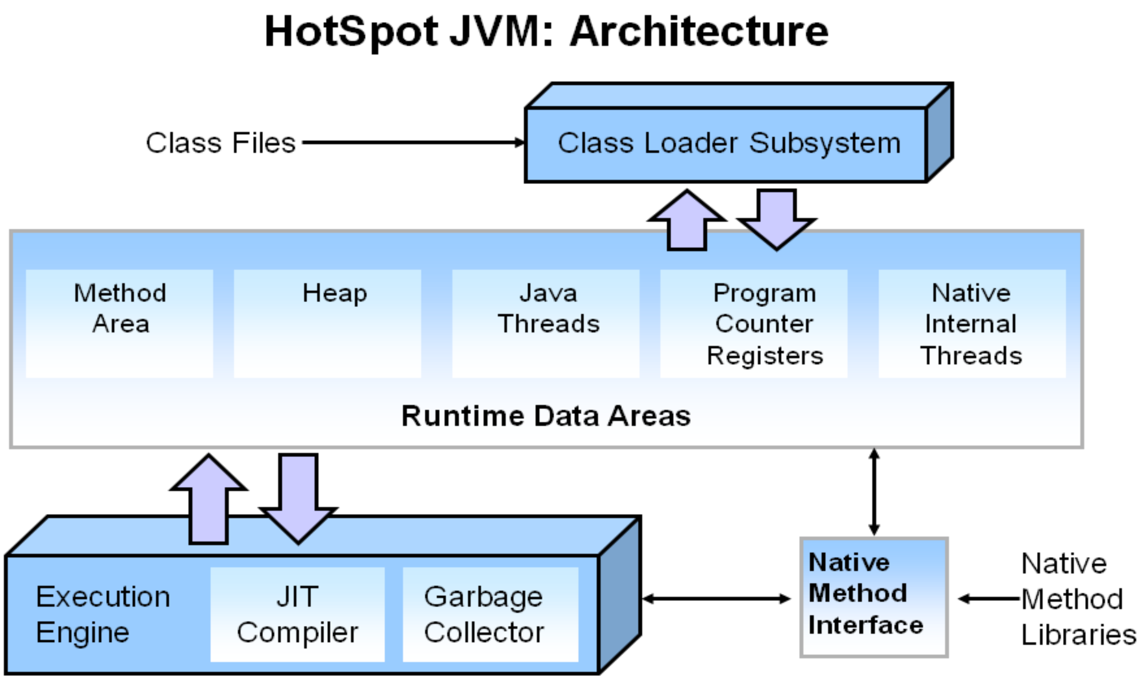
## 25. .what is diff b/w String str1="xyz"; and String str2= new String ("xyz");

Answer:

In String str2=new String(“XYZ”); in this statement 2 string object will be created one is for **new** keyword so it’s reference will be stored in heap and another one is for String literal “XYZ” and it’s reference will be stored in SCPA

In String str1=”XYZ”; here only one object will be created and reference will be stored in SCPA if that string object is not available in SCPA before , if it is already exist in SCPA than simply it will return the same instance.

### 26. Explain about JVM architecture?



1. When we run the command javac demo.java then .class file will be generated and then JVM request to class loader to load the class in method area.

2. All the static and instance field will be loaded to method area with default value

3. All the object creation part will be done in heap area and reference will be stored in heap

4. Normally initially one default thread is available in runtime so if I create multiple threads then it allocates memory in Thread area.

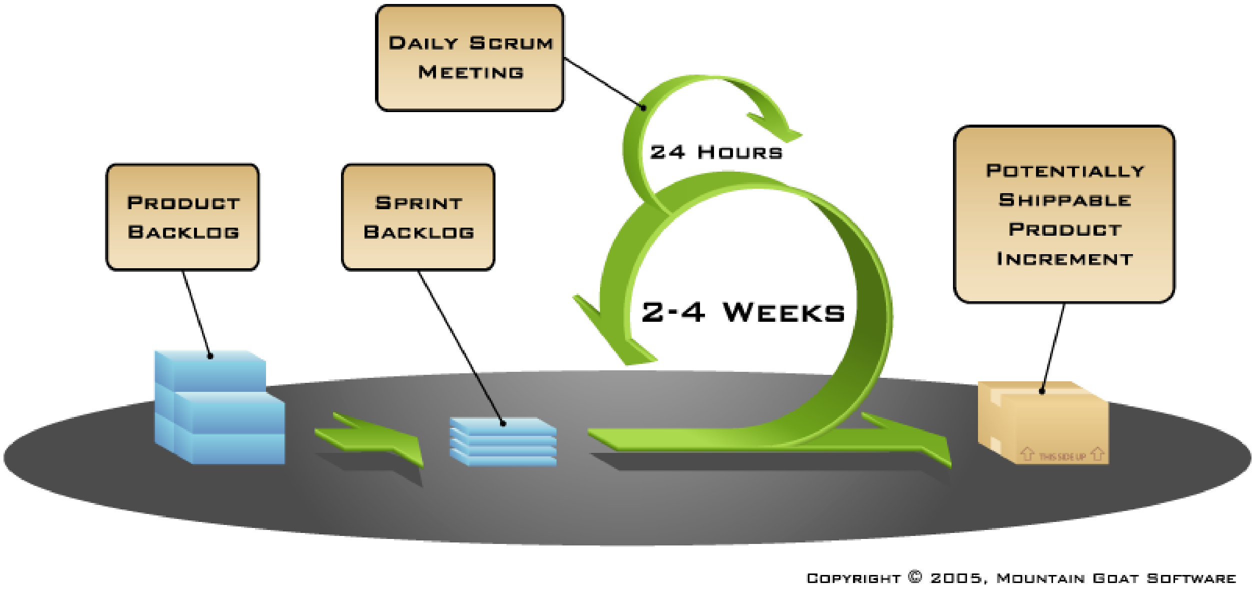
5. When we call a method then it allocate memory in Java Stack area and internally create one stack frame and after execution of method stack frame will be destroy.

6. Garbage collector is there to destroy unused objects

## 27. Difference between SOAP and Rest?

| SOAP | REST |
|---|---|
| 1.In SOAP base web service we need to depends on SOPA protocol | 1.Here protocol can be anything |
| 2.Here message format should be XML | 2.Here Message format can be anything like XML,JSON,PLAIN |
| 3.inbuild Cache mechanism is not there | 3.Inbuild Cache mechanism is available |
| 4.For security purpose SOAP is preferable | 4.For quick access REST is preferable |
| 5.To develop SOAP base web Service is little bit complicated cause we have to run the tool generate binding class get the stub call the service endpoint method | 5.In Rest base simple get the URI and pass the input data get the response ,which is easier than SOAP approach |

## 28. Agile methodology?


Daily Scrum Meeting
24 Hours
Product Backlog
Sprint Backlog
2-4 Weeks
Potentially Shippable Product Increment
Copyright © 2005, Mountain Goat Software

1. Normally Business requirement gather by Business Analyst from Client, complete requirements will put in Product Backlog which acts as a container who stores the all business requirements.

2. From this complete Requirements gather the functionality and then make a specific requirement as a one sprint means simple each module we can say one sprint which will be stored in Sprint Backlog.

3. We have to pull one sub-module from sprint Backlog and as per documentation we have to developed the code to full fill business requirement

If we will stuck in any situation first we have to try to resolve it in offshore if not solved then we have to update the status in daily order scrum meeting then we will get multiple approach to resolve this issue

4.Every sprint have some time duration which will be estimate by scrum master and the duration max to 2-3 weeks and monthly one time we have to discuss with onsite team in presents of scrum master regarding spring which is called as Hands-on meeting

5. The process of development is going in cyclic manner if completed then put it in Sprint repository for next phase. Again process will be continuing…

## 29. How to create web-services project and spring project using maven?

Simply we have to create one maven project like maven-archetype-webapp then add the dependency from local repository if available else download from central repository in pom.xml.

Add spring dependency along with JAX-WS implementation class dependency in pom.xml

## 30. What is diff b/w throw and throws?

Throw:

Normally Throw keyword is used to throw exception from class means if I want to throw a custom exception then I have to manually throw it from method level it can be checked or unchecked . if you throw exception we have to handle it either by using try catch or throws keyword

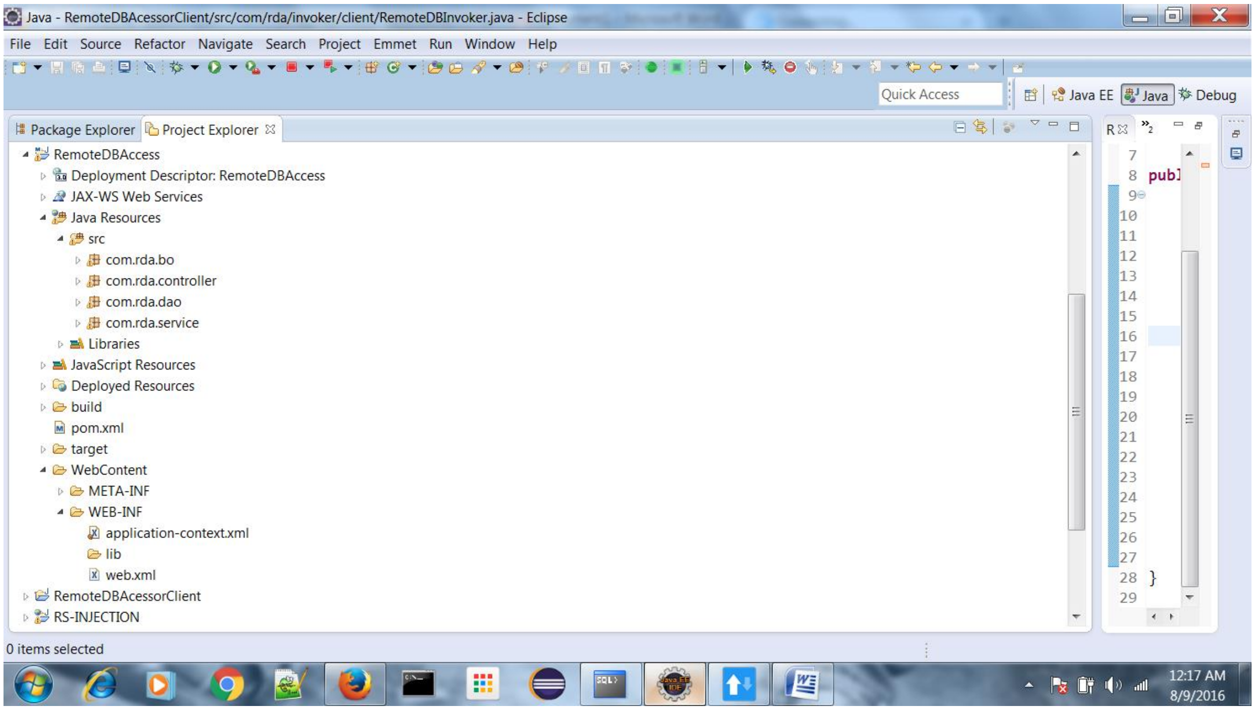
Throws:

Throws Keyword used to intimate the method caller regarding exception and it can be applicable in method level then when other method try to call your method he will know about exception by seeing the syntax of throws, if the exception is checked then better to prefer throws keyword

### 31. I have a company table in remote database. By using rest I need to get the table data and print into a file?

# Access Remote Database through Rest API

## Resource Part:



## BO (Model class)

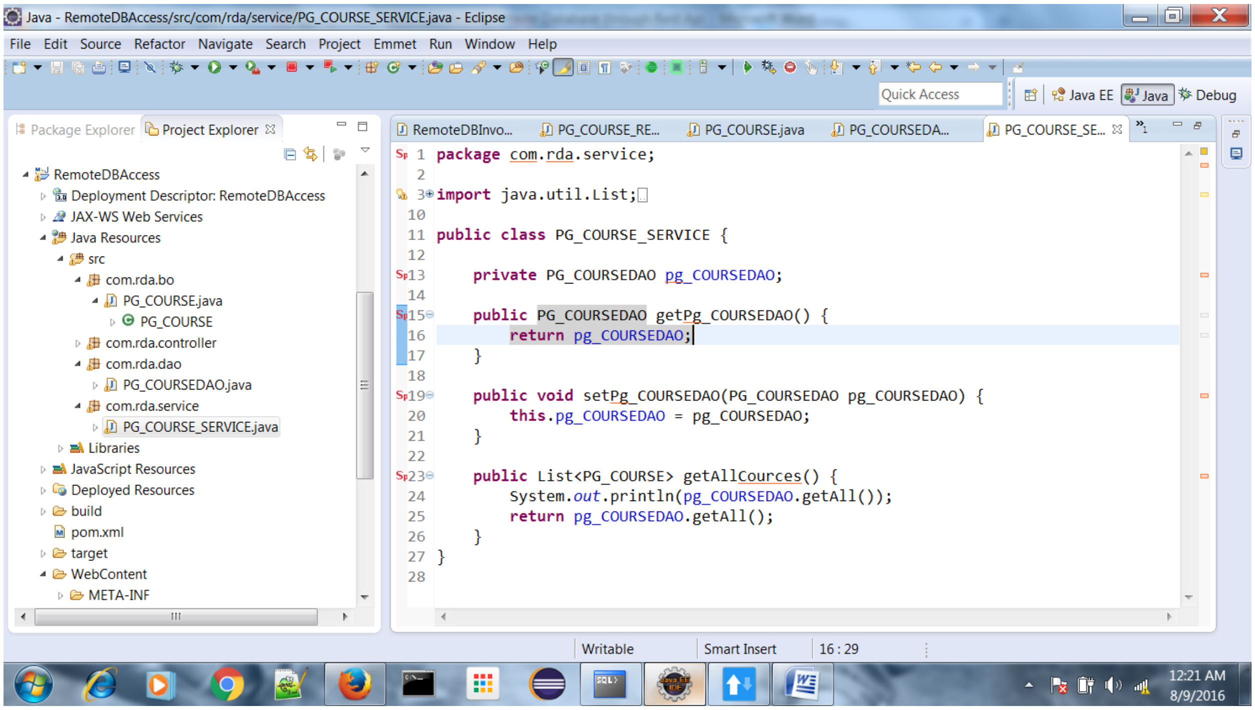
```
package com.rda.bo;

import java.io.Serializable;

public class PG_COURSE implements Serializable{
	private int courseId;
	private String courseName;

	public int getCourseId() {
		return courseId;
	}

	public void setCourseId(int courseId) {
		this.courseId = courseId;
	}

	public String getCourseName() {
		return courseName;
	}
	public void setCourseName(String courseName) {
		this.courseName = courseName;
	}

	public PG_COURSE(int courseId, String courseName) {
		this.courseId = courseId;
		this.courseName = courseName;
	}

	public PG_COURSE() {

	}

	@Override
	public String toString() {
		return "PG_COURSE [courseId=" + courseId + ", courseName=" + courseName
				+ "]";
	}

}
```

## DAO

```
package com.rda.dao;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.jdbc.core.RowMapper;

import org.springframework.stereotype.Repository;

import com.rda.bo.PG_COURSE;

public class PG_COURSEDAO {

	private JdbcTemplate jdbcTemplate;

	public JdbcTemplate getJdbcTemplate() {

		return jdbcTemplate;

	}



	public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {

		this.jdbcTemplate = jdbcTemplate;
```

```
    }

    final String SQL_QUERY_FOR_READ_ALL_RECORDS = "SELECT COURSE_ID,COURSE_NAME FROM PG_COURSE";

    public List<PG_COURSE> getAll() {

        return jdbcTemplate.query(SQL_QUERY_FOR_READ_ALL_RECORDS,

                new RowMapper<PG_COURSE>() {


                    @Override

                    public PG_COURSE mapRow(ResultSet rs, int arg1)

                            throws SQLException {

                        return new PG_COURSE(rs.getInt("COURSE_ID"), rs

                                .getString("COURSE_NAME"));

                    }

                });

    }

}
```

## Service:



```
package com.rda.service;

import java.util.List;

public class PG_COURSE_SERVICE {

    private PG_COURSEDAO pg_COURSEDAO;

    public PG_COURSEDAO getPg_COURSEDAO() {
        return pg_COURSEDAO;
    }

    public void setPg_COURSEDAO(PG_COURSEDAO pg_COURSEDAO) {
        this.pg_COURSEDAO = pg_COURSEDAO;
    }

    public List<PG_COURSE> getAllCources() {
        System.out.println(pg_COURSEDAO.getAll());
        return pg_COURSEDAO.getAll();
    }
}
```

## Controller

**package com.rda.controller;**

**import java.io.File;**

**import java.io.FileOutputStream;**

**import java.io.IOException;**

**import java.io.ObjectOutputStream;**

**import java.util.List;**

**import javax.ws.rs.GET;**

**import javax.ws.rs.Path;**

**import javax.ws.rs.Produces;**

**import javax.ws.rs.core.MediaType;**

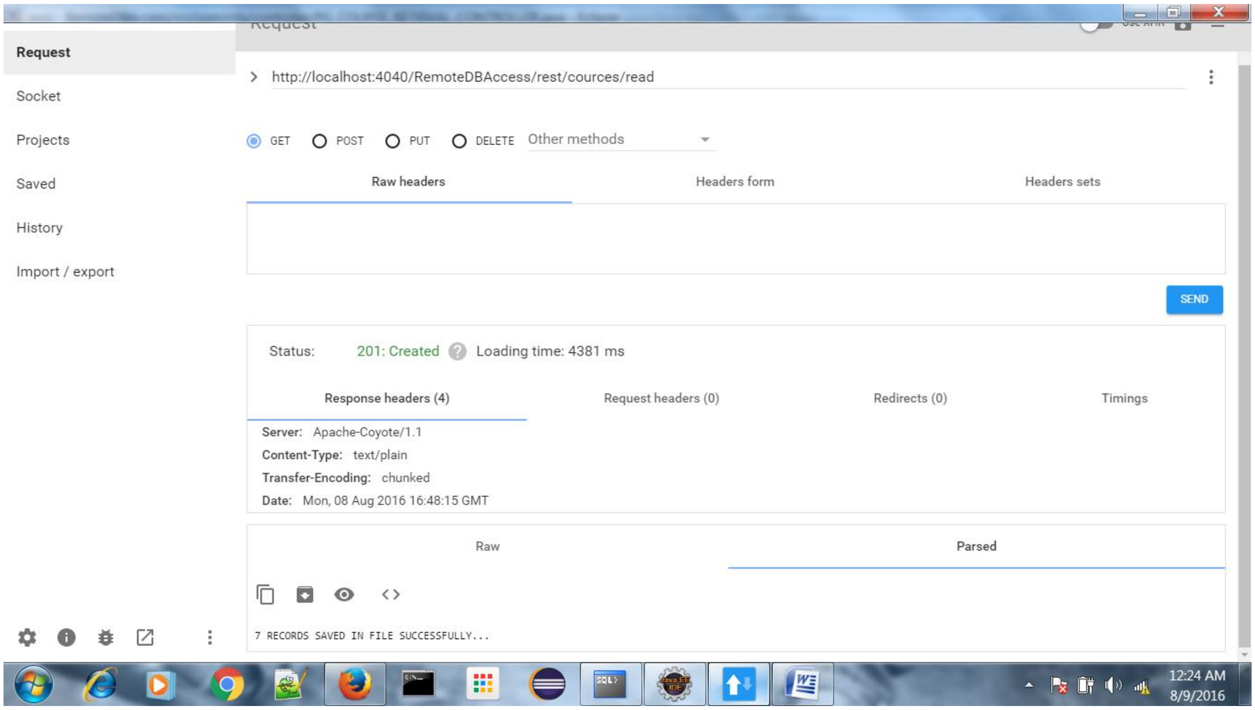
**import javax.ws.rs.core.Response;**

**import com.rda.bo.PG_COURSE;**

**import com.rda.service.PG_COURSE_SERVICE;**

```
@Path("/cources")
public class PG_COURSE_RETRIVAL_CONTROLLER {
    private PG_COURSE_SERVICE service;
    public PG_COURSE_SERVICE getService() {
        return service;
    }
    public void setService(PG_COURSE_SERVICE service) {
        this.service = service;
    }
    @Path("/read")
    @GET
    @Produces(MediaType.TEXT_PLAIN)
    public Response readAllCources() throws IOException {
        int count = 0;
        String message = "";
        List<PG_COURSE> courses = service.getAllCources();

        FileOutputStream fos = new FileOutputStream(new File(
                "D:\\DOCUMENT\\remotData.txt"));
        ObjectOutputStream oos = new ObjectOutputStream(fos);
        for (PG_COURSE course : courses) {
            count++;
            oos.writeObject(course);
        }
        message = count + " RECORDS SAVED IN FILE SUCCESSFULLY...";
        return Response.ok().entity(message).status(201).build();
    }
}
```

# ARCT(Advance Rest Client)



http://localhost:4040/RemoteDBAccess/rest/cources/read

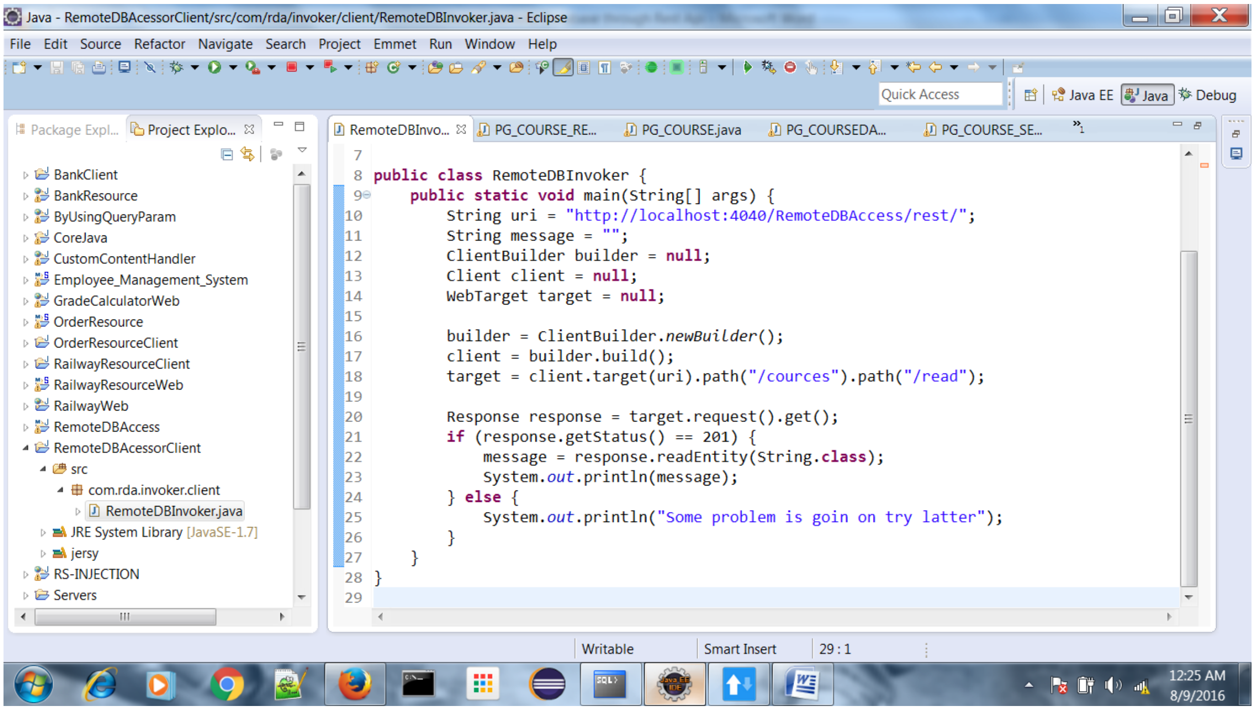
● GET ○ POST ○ PUT ○ DELETE

Status: 201: Created Loading time: 4381 ms

Server: Apache-Coyote/1.1
Content-Type: text/plain
Transfer-Encoding: chunked
Date: Mon, 08 Aug 2016 16:48:15 GMT

```
7 RECORDS SAVED IN FILE SUCCESSFULLY...
```

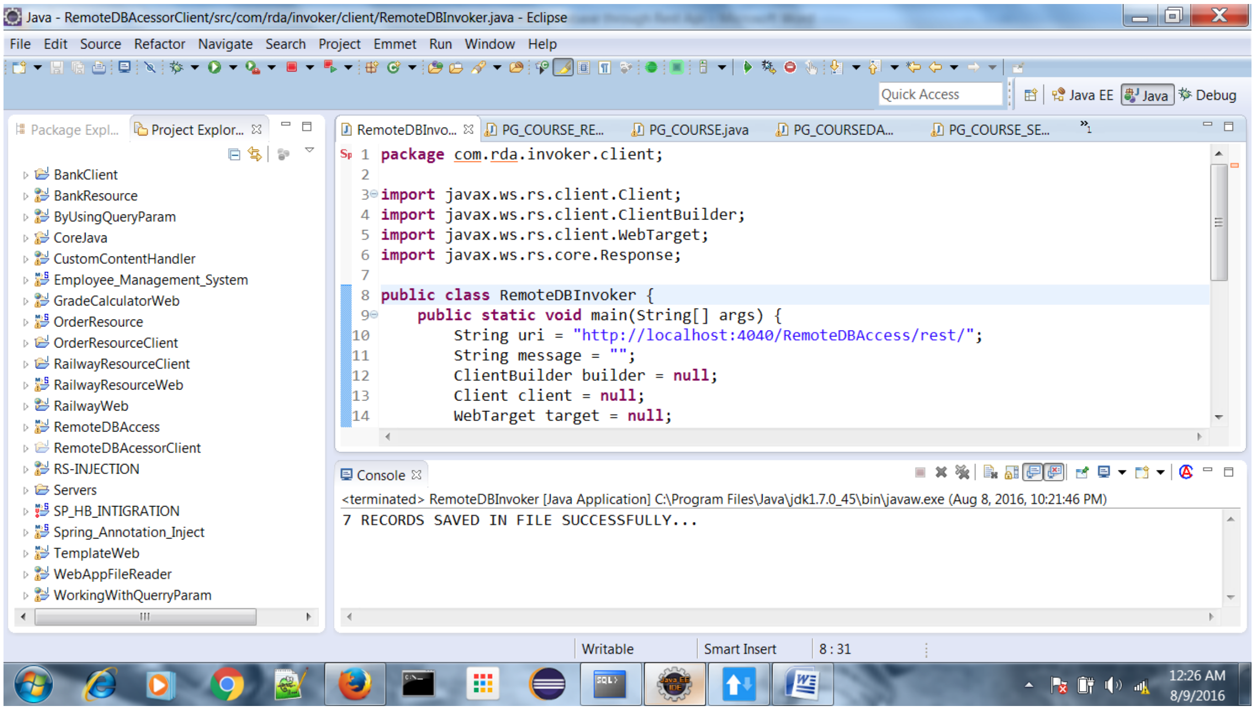
# Invoker CLIENT PART



Java - RemoteDBAcessorClient/src/com/rda/invoker/client/RemoteDBInvoker.java - Eclipse

```java

public class RemoteDBInvoker {
    public static void main(String[] args) {
        String uri = "http://localhost:4040/RemoteDBAccess/rest/";
        String message = "";
        ClientBuilder builder = null;
        Client client = null;
        WebTarget target = null;

        builder = ClientBuilder.newBuilder();
        client = builder.build();
        target = client.target(uri).path("/cources").path("/read");

        Response response = target.request().get();
        if (response.getStatus() == 201) {
            message = response.readEntity(String.class);
            System.out.println(message);
        } else {
            System.out.println("Some problem is goin on try latter");
        }
    }
}

```

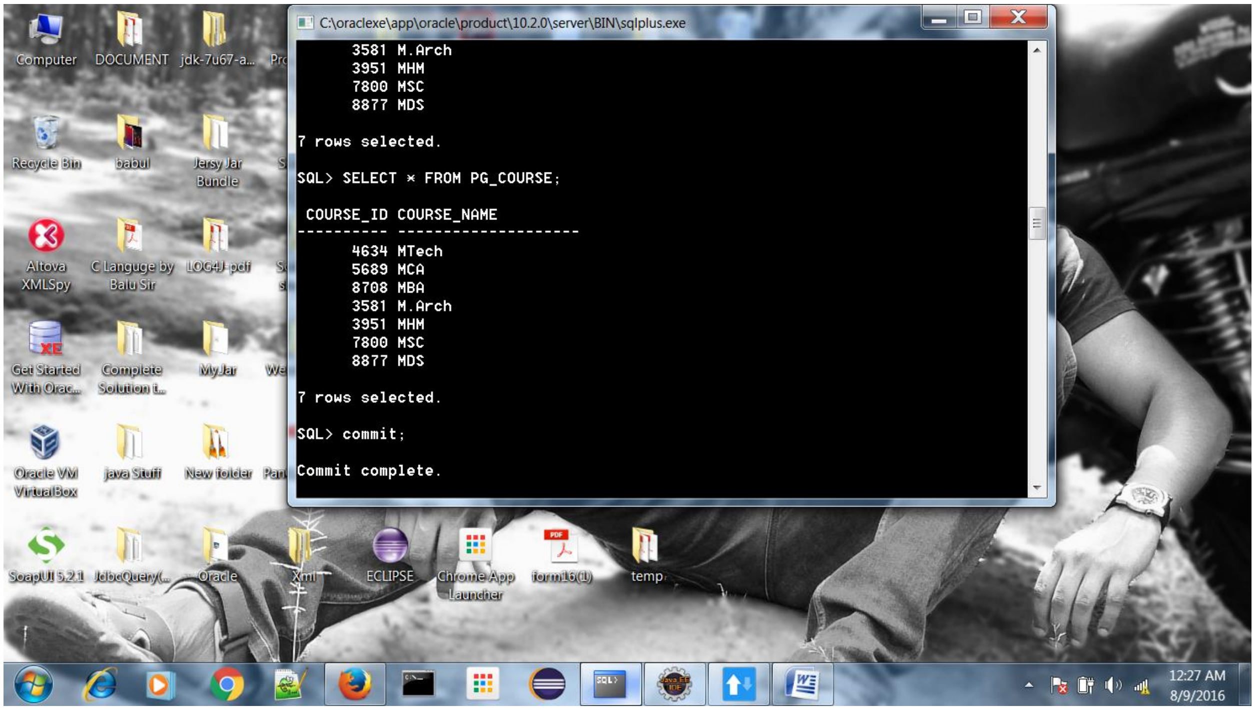
# Response by client API:



```java
package com.rda.invoker.client;

import javax.ws.rs.client.Client;
import javax.ws.rs.client.ClientBuilder;
import javax.ws.rs.client.WebTarget;
import javax.ws.rs.core.Response;

public class RemoteDBInvoker {
    public static void main(String[] args) {
        String uri = "http://localhost:4040/RemoteDBAccess/rest/";
        String message = "";
        ClientBuilder builder = null;
        Client client = null;
        WebTarget target = null;
```

```
<terminated> RemoteDBInvoker [Java Application] C:\Program Files\Java\jdk1.7.0_45\bin\javaw.exe (Aug 8, 2016, 10:21:46 PM)
7 RECORDS SAVED IN FILE SUCCESSFULLY...
```

# Database:



```
      3581 M.Arch
      3951 MHM
      7800 MSC
      8877 MDS

7 rows selected.

SQL> SELECT * FROM PG_COURSE;

 COURSE_ID COURSE_NAME
---------- --------------------
      4634 MTech
      5689 MCA
      8708 MBA
      3581 M.Arch
      3951 MHM
      7800 MSC
      8877 MDS

7 rows selected.

SQL> commit;

Commit complete.
```

## 32. Can you tell me java8 features?

There is lot of enhancement done in Java 8 like

1. Lambda Expression

2. Default method

3. Functional interface

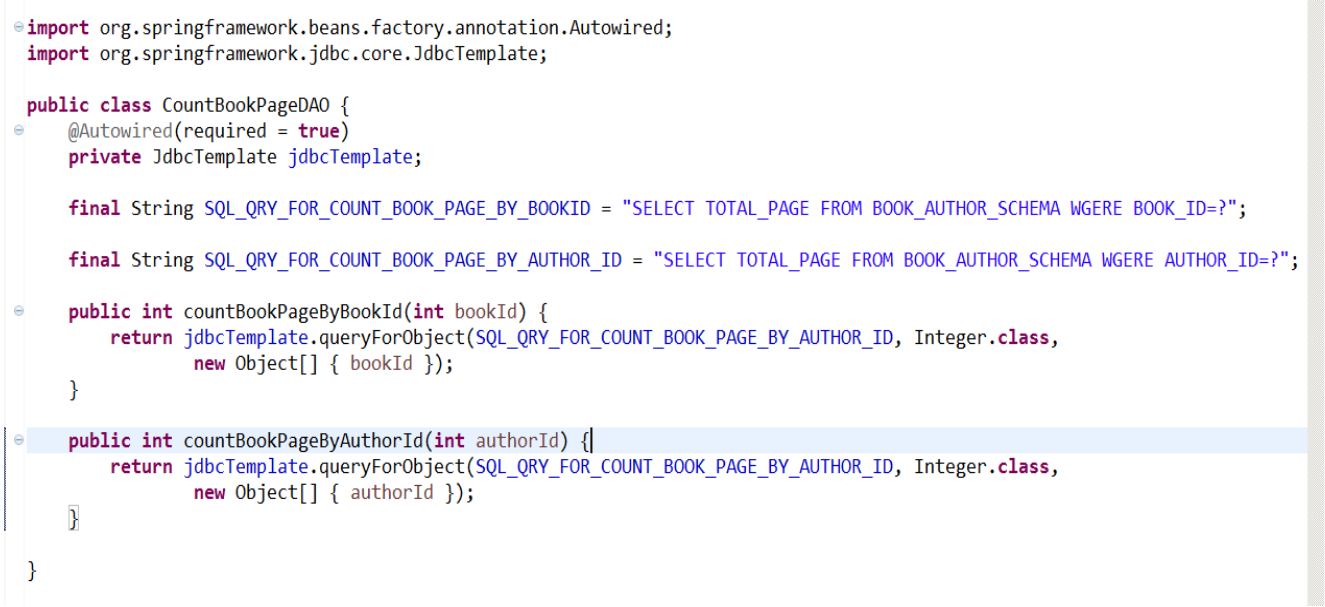
4. Method Reference etc…

## 33. How to read book pages on online library by using bookid or author id(by using restful services)?

Just assume my schema structure is as below

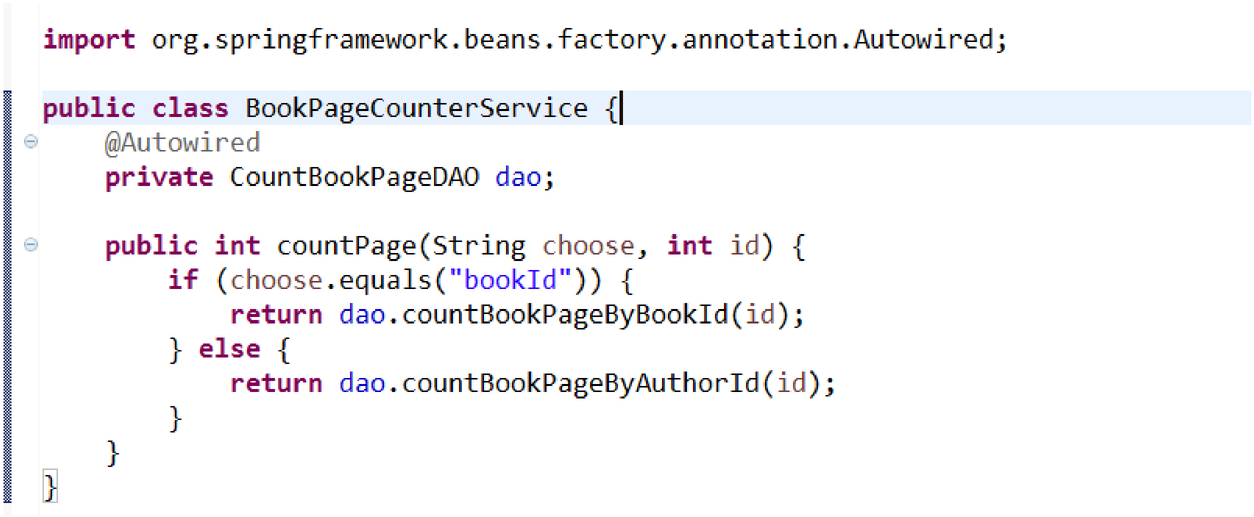
| SL.NO | BOOK_ID | TOTAL_PAGE | PUBLISH_DATE | AUTHOR_ID |
|---|---|---|---|---|
| 1374897 | 234 | 4209 | 08.FEB.2014 | 462 |
| 3462328 | 253 | 708 | 12.MAR.2012 | 346 |
| 7573457 | 123 | 2411 | 01.JAN.2011 | 356 |

**Write DAO:**



```java
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.jdbc.core.JdbcTemplate;

public class CountBookPageDAO {
    @Autowired(required = true)
    private JdbcTemplate jdbcTemplate;

    final String SQL_QRY_FOR_COUNT_BOOK_PAGE_BY_BOOKID = "SELECT TOTAL_PAGE FROM BOOK_AUTHOR_SCHEMA WGERE BOOK_ID=?";

    final String SQL_QRY_FOR_COUNT_BOOK_PAGE_BY_AUTHOR_ID = "SELECT TOTAL_PAGE FROM BOOK_AUTHOR_SCHEMA WGERE AUTHOR_ID=?";

    public int countBookPageByBookId(int bookId) {
        return jdbcTemplate.queryForObject(SQL_QRY_FOR_COUNT_BOOK_PAGE_BY_AUTHOR_ID, Integer.class,
                new Object[] { bookId });
    }

    public int countBookPageByAuthorId(int authorId) {
        return jdbcTemplate.queryForObject(SQL_QRY_FOR_COUNT_BOOK_PAGE_BY_AUTHOR_ID, Integer.class,
                new Object[] { authorId });
    }

}
```

**Write Service:**



```
import org.springframework.beans.factory.annotation.Autowired;

public class BookPageCounterService {
    @Autowired
    private CountBookPageDAO dao;

    public int countPage(String choose, int id) {
        if (choose.equals("bookId")) {
            return dao.countBookPageByBookId(id);
        } else {
            return dao.countBookPageByAuthorId(id);
        }
    }
}
```

**Write Recourse class:**

```
@Path("/book")
public class BookPageCountResource {
      private BookPageCounterService service;

      @Path("/{option}/{id}")
       @GET
      public Response getCount(@PathParm ("option") String option, @PathParm("id")
int id) {
             int totalPage = service.countPage(option, id);
             String message = "Total no of page available is :" + totalPage + " for
this" + option + "=" + id;
             return Response.ok().entity(message).build();
      }

}
```

**NOTE**: Access it throw Rest otherwise write client class .already I write one client class in above example follow the same to develop client

## 34. I have a table in remote database, how to update the data in that table using rest?

Simple write the update logic in DAO and call this from controller class but in controller class u have to mention the annotation @PUT for update through network. Code try as per above 2 example only logic will be changed but not approach.

## 35. What are all the contents in WSDL?

WSDL act as a contract between consumer and provider so all the provider details we have to mention in WSDL otherwise a consumer can't know the details of Business requirements. That's why it contains multiple sections and each section specifies the separate role.

1. Definitions

2. Types (talk about required input and output in details)

3. Messages (talks about all required input and output in single entity)

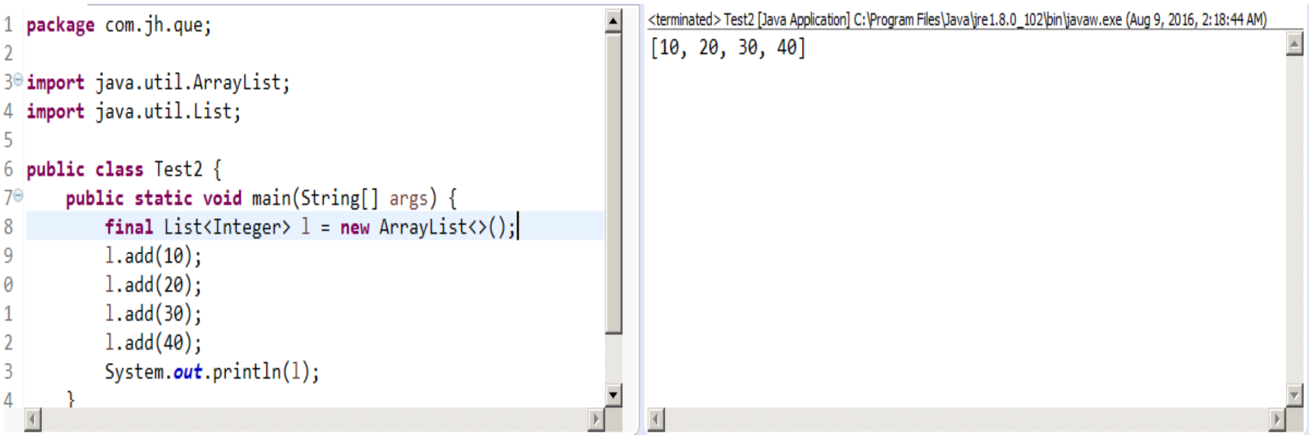
4. PortType (it gives the complete structure of Service end point interface)

5. Binding (it talks about the MEF and protocol details)

6. Service (It gives the URL address and location)

## 36. Can I add elements to list, if it is defined as final?

Yes we can add object to the list if it is final also as below

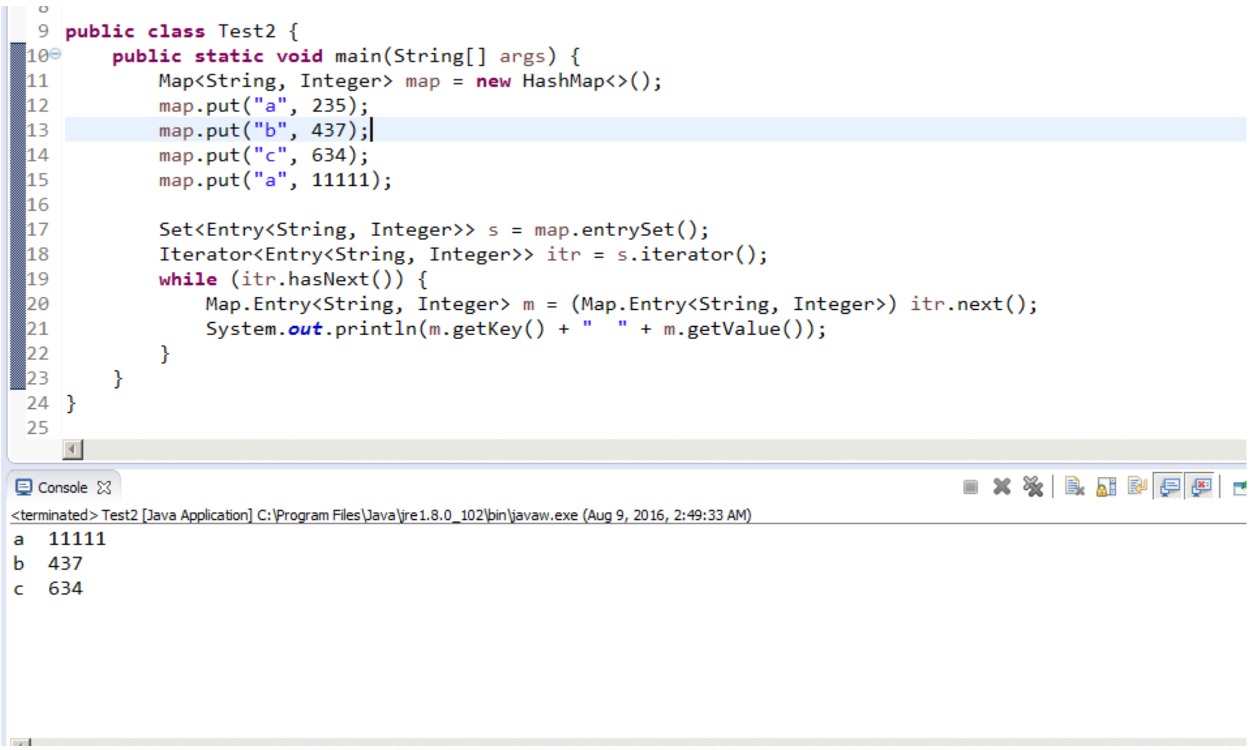


```java
package com.jh.que;

import java.util.ArrayList;
import java.util.List;

public class Test2 {
    public static void main(String[] args) {
        final List<Integer> l = new ArrayList<>();
        l.add(10);
        l.add(20);
        l.add(30);
        l.add(40);
        System.out.println(l);
    }
```

```
<terminated> Test2 [Java Application] C:\Program Files\Java\jre1.8.0_102\bin\javaw.exe (Aug 9, 2016, 2:18:44 AM)
[10, 20, 30, 40]
```

## 37. .if you pass duplicate key to map what will happen?

If I pass duplicate key in Map then key collision will be occur and the new pass value will be replaced in existing key. See the below example u will got clear cut idea.



```
public class Test2 {
    public static void main(String[] args) {
        Map<String, Integer> map = new HashMap<>();
        map.put("a", 235);
        map.put("b", 437);
        map.put("c", 634);
        map.put("a", 11111);

        Set<Entry<String, Integer>> s = map.entrySet();
        Iterator<Entry<String, Integer>> itr = s.iterator();
        while (itr.hasNext()) {
            Map.Entry<String, Integer> m = (Map.Entry<String, Integer>) itr.next();
            System.out.println(m.getKey() + "  " + m.getValue());
        }
    }
}
```

```
<terminated> Test2 [Java Application] C:\Program Files\Java\jre1.8.0_102\bin\javaw.exe (Aug 9, 2016, 2:49:33 AM)
a  11111
b  437
c  634
```

In this above programmed I pass new value is 11111 and old value is 235 so new value will be replaced in existing value

## 38. Difference between Abstract class and interface?

| Abstract class | Interface |
|---|---|
| 1.Abstract class provide partially implementation | 1.Interface is fully un implemented class |
| 2.It make our class as partially decoupled | 2.It make our class as fully decoupled |
| 3.We can achieve partial Runtime polymorphism | 3.We can achieve fully Runtime polymorphism |
| 4.Fully abstraction is not there | 4.we can achieve fully abstraction |
| 5.it contain both abstract method and concrete method | 5.It contains only abstract method |
| 6.Abstract keyword will be used in class level and method level | 6.No keyword required |

## 39. Diff b/w comparator and comparable?

| Comparable | comparator |
|---|---|
| 1.it used for natural sorting | 1.it used for custom sorting |
| 2.Present in java.lang package so it can be applicable for any class | 2.Present in java,util.*; package |
| 3.Comparable provide compareTo() method to sort element | 3.Comparator provide compare() method to sort element |

## 40. How to set timeout for the browser? (Clue: restful client API)?

```
Client client = ClientBuilder.newClient();

    client.property(ClientProperties.CONNECT_TIMEOUT, 1000);
    client.property(ClientProperties.READ_TIMEOUT,    1000);
```

## 41. In written test they are asking sorting programs (bubble sort, quicksort)

### Bubble-Sort:

```
package com.core.all.interview.programmes;

public class SortArrayByBubbleSort {

      public static void sort(int input[]) {
            int temp = 0;
            for (int i = 0; i <= input.length - 1; i++) {
                  for (int j = i + 1; j <= input.length - 1; j++) {
                        if (input[i] > input[j]) {
                              temp = input[i];
                              input[i] = input[j];
                              input[j] = temp;
                        }
                  }
                  System.out.println(input[i]);
            }
      }

      public static void main(String[] args) {
            int[] i = new int[] { 12, 44, 23, 43, 21, 8, 0, 6, 45, 44, 58, 17
};
```

```
			sort(i);
		}
	}
```

### Quicksort:

```
package com.core.all.interview.programmes;

import java.util.Arrays;

public class SortArrayByQuickSort {

	public static void quickSort(int[] arr, int low, int high) {
		if (arr == null || arr.length == 0)
			return;

		if (low >= high)
			return;

		int middle = low + (high - low) / 2;
		int pivot = arr[middle];

		// make left < pivot and right > pivot

		int i = low, j = high;
		while (i <= j) {
			while (arr[i] < pivot) {
				i++;
			}

			while (arr[j] > pivot) {
				j--;
			}

			if (i <= j) {
				int temp = arr[i];
				arr[i] = arr[j];
				arr[j] = temp;
				i++;
				j--;
			}
		}

		// recursively sort two sub parts

		if (low < j)
			quickSort(arr, low, j);

		if (high > i)
			quickSort(arr, i, high);
```

```
    }

    public static void main(String[] args) {
        int[] i = new int[] { 12, 44, 23, 43, 21, 8, 0, 6, 45, 44, 58, 17
};
        quickSort(i, 0, i.length - 1);
        System.out.println(Arrays.toString(i));
    }
}
```

## 42. What is time complexity? If you are going to implement sorting by your own which sorting you prefer? And why?

If we are writing logic for custom sorting then we will get better performance than predefined ,I will prefer to use Bubble sort where time complexity is less and shifting management will not require manually .

## 43. Use of volatile and synchronized keyword?

### *Synchronized:*

When multiple threads try to access same object at a time then there may be data conflict to avoid this we have to use synchronized keyword to avoid data consistency

### *Volatile:*

When multiple thread try to access same object and one thread modified the property of object then that modification will not visible to other thread cause the thread who modified he caching the data so all modification done by thread will be stored in own cache, so if we want thread will execute without caching data then we have to use volatile keyword.

## 44. What is serialization? Have you implement serialization in your project?

Serialization is a process where we can change state of object to the file over the network or simply we can transfer our object from one layer to another layer, that’s why java provides streaming API.

In my project I used in pojo class means my pojo class should be implements from Serializable interface because that business object will be transfer over the network, that’s why it’s recommended to implements BO object from Serializable interface.

## 45. Why we are using @qualifier?

@Qualifier annotation is used to avoid ambiguity error , simple in spring bean configuration file when I configure same class type with different id then IOC get confuse which bean he should have to inject so to overcome this problem spring framework provide @Qualifier annotation like

@Qualifier (“bean id”) whose id u will pass here only that bean will be injecting by IOC.

## 46. Difference between Bean Factory and Application Context?

| ***Application Context*** | ***Bean Factory*** |
|---|---|
| 1.Application Context is eager initializer | 1.Bean Factory is lazy initializer |
| 2.I18N support by Application Context | 2.I18N not supported |
| 3.No need to manually register BeanPostProcessor or BeanFactoryPostProcessor | 3.We need to manually register in IOC about BeanPostProcessor or BeanFactoryPostProcessor |
| 4.Even Driven base supported | 4. Even Driven base not supported |
| 5.Application Context is sub interface of Bean Factory | 5.Bean Factory is super interface of Application Context |

## 47. Explain about IOC container?

IOC container means simple a class who provide runtime support to our spring beans. To instantiate IOC container 2 interface provide by spring framework 1 is BeanFactory another one is ApplicationContext so after instantiate it internally create in memory Meta data where all the spring bean details will be stored and then dependency injection, component/Resource management was done by IOC itself.

## 48. How you are implemented polymorphism in your project?

Normally we are developing our application interface base approach for example service and serviceImpl so my controller need to inject serviceImpl to get business functionality so in this case in controller class we are injection service bean by taking service interface reference as below .

```
public interface BankService {
	public void doTransaction();
}


@Service
public class BankServiceImpl implements BankService{
	@Override
	public void doTransaction() {
		// LOGIC
	}
}

@Controller
public class BankController {
	@Autowired(required = true)
	private BankService service;
}
```

Here internally IOC container instantiate my service bean as below approach

```
BankService service=new BankServiceImpl();//Runtime polymorphism
```

## 49. Diff b/w Iterator and List Iterator and Enumerator?

Simple the basic difference

Iterator is applicable for all collection implemented class like List, Set or Map or In iterator we can moves our cursor only in forward direction

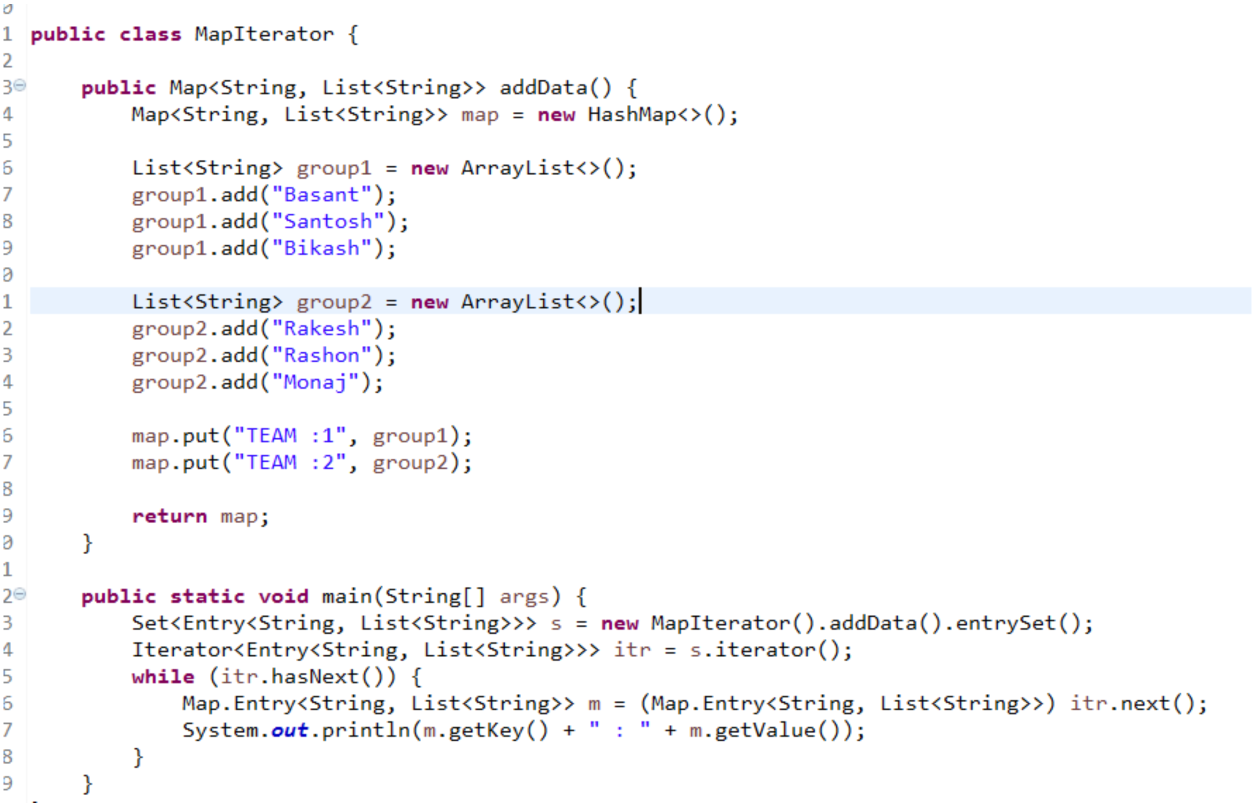
List Iterator is applicable for only List implemented class like ArrayList & LinkedList In List iterator we can moves our cursor in both direction like forward and backward

Enumerator is applicable for legacy implemented class like Vector, stack etc In Enumerator we can moves our cursor only in forward direction

## 50. What are all the collections are supporting ListIterator?

List Iterator is applicable for only List implemented class like ArrayList , LinkedList and vector also.

## 51. How you iterate map having key and list<values>?



```
public class MapIterator {

    public Map<String, List<String>> addData() {
        Map<String, List<String>> map = new HashMap<>();

        List<String> group1 = new ArrayList<>();
        group1.add("Basant");
        group1.add("Santosh");
        group1.add("Bikash");

        List<String> group2 = new ArrayList<>();
        group2.add("Rakesh");
        group2.add("Rashon");
        group2.add("Monaj");

        map.put("TEAM :1", group1);
        map.put("TEAM :2", group2);

        return map;
    }

    public static void main(String[] args) {
        Set<Entry<String, List<String>>> s = new MapIterator().addData().entrySet();
        Iterator<Entry<String, List<String>>> itr = s.iterator();
        while (itr.hasNext()) {
            Map.Entry<String, List<String>> m = (Map.Entry<String, List<String>>) itr.next();
            System.out.println(m.getKey() + " : " + m.getValue());
        }
    }
```

## 52. How to make non-synchronized map and list as synchronized (by using collection method)?

Synchronized Map:

Map m=Collections.synchronizedMap(map);

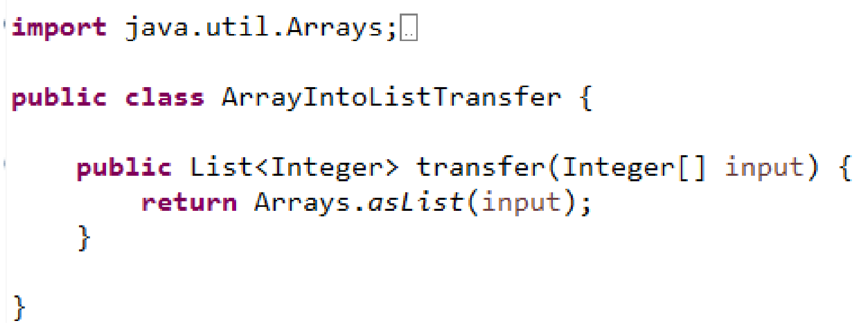
Synchronized List:

List l=Collections.synchronizedList(list);

## 53. .what is diff b/w collection and collections?

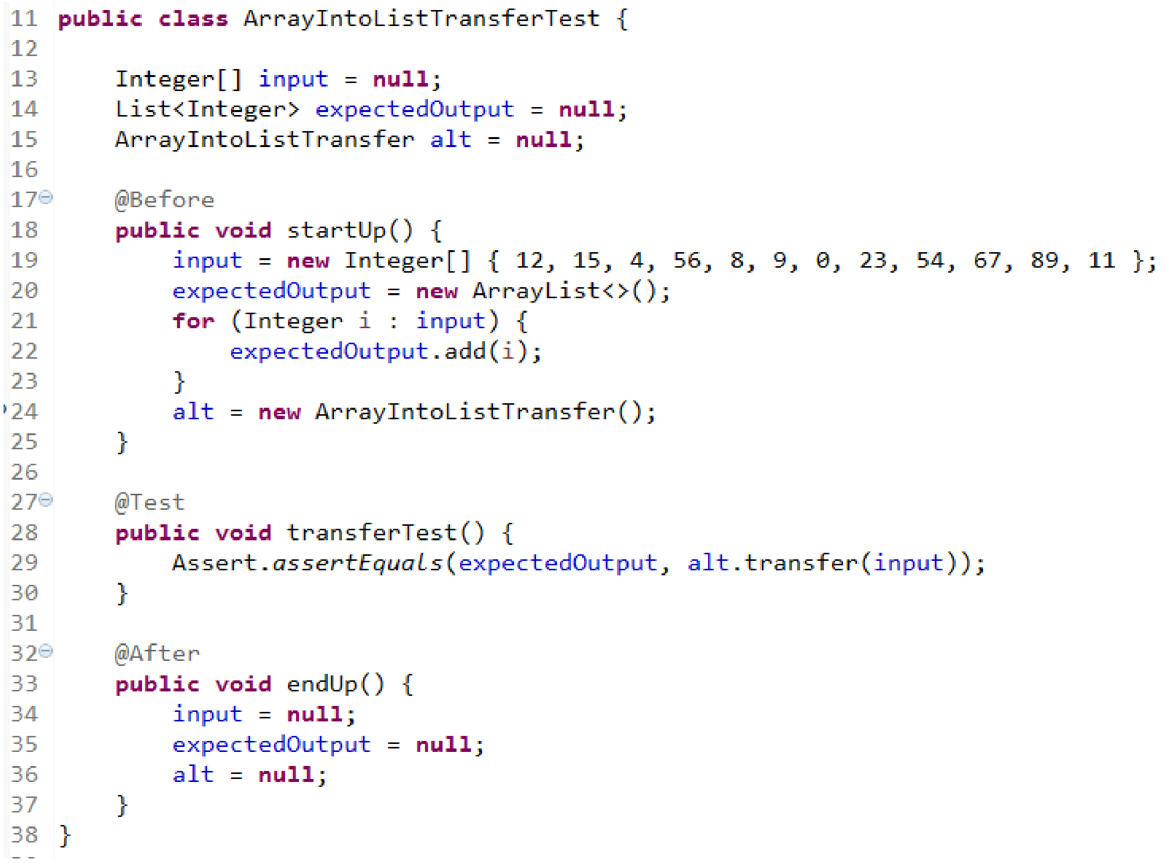
Collection is an interface which provide multiple sub implementation for grouping/Storing heterogeneous and homogeneous object in various manner like insertion, sorting where Collections is an class who contain multiple predefined method to make work easy for developer like sort, reverse, binary search etc.

## 53. Write the junit test case for the below scenario? Read array of elements into list<String>.

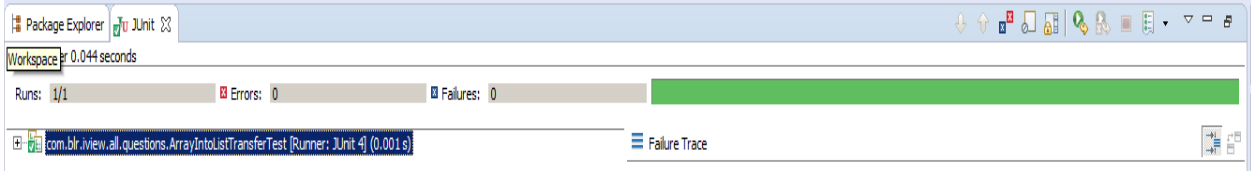


```
import java.util.Arrays;

public class ArrayIntoListTransfer {

    public List<Integer> transfer(Integer[] input) {
        return Arrays.asList(input);
    }

}
```

## Test Case:



```
public class ArrayIntoListTransferTest {

    Integer[] input = null;
    List<Integer> expectedOutput = null;
    ArrayIntoListTransfer alt = null;

    @Before
    public void startUp() {
        input = new Integer[] { 12, 15, 4, 56, 8, 9, 0, 23, 54, 67, 89, 11 };
        expectedOutput = new ArrayList<>();
        for (Integer i : input) {
            expectedOutput.add(i);
        }
        alt = new ArrayIntoListTransfer();
    }

    @Test
    public void transferTest() {
        Assert.assertEquals(expectedOutput, alt.transfer(input));
    }

    @After
    public void endUp() {
        input = null;
        expectedOutput = null;
        alt = null;
    }
}
```



```
Workspace r 0.044 seconds
Runs: 1/1    Errors: 0    Failures: 0
com.blr.iview.all.questions.ArrayIntoListTransferTest [Runner: JUnit 4] (0.001 s)
```

## 54. What is diff b/w spring-jdbc and hibernate?

There is no more different because spring provided integration with both JDBC and Hibernate but based on requirement we have to choose,

If my application is internet application with having multiple users so here data operation is more so we have to choose JDBC approach. Ex: Banking Application

If my application is intranet application where the access user is not more and data operation is less then in this case we have to use hibernate Ex: Any internal application for specific organization.

## 55. What are all the drawbacks of jdbc over hibernate?

The basic drawbacks are:

1. In case of JDBC we need to do all resource management manually like close the connection, Statement, Resultset here in Hibernate Resource management will done by ORM itself.

2. In case of JDBC we need to do Exception Mapping at the time of closing statement but in Hibernate it's not required.

3. Caching mechanism is not available in JDBC, which is available in Hibernate.

4. JDBC is Database Dependent where Hibernate is Database independent

5. Connection Reusability is not there in JDBC.

## 56. What are all the problems with inheritance?

Normally by using inheritance (IS-A) my class tightly coupled with super logical class name

- Multiple inheritance not supported
- We can achieve partial Runtime polymorphism cause super method is concert not abstract
- We can achieve partially abstraction

## 57. What is the use of Hibernate session?

In Hibernate Session is act as lead role who provide all persistent predefined method to perform database operation that’s why to get session implemented object we need to call one method i.e. openSession() which is instance factory method presents in Session Factory interface

The session objects should not be kept open for a long time because they are not usually thread safe and they should be created and destroyed them as needed.

## 58. Why we are using @transient in hibernate?

@Transient annotation is used for avoid to make property as a column in Database for example.

```
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Table;
import javax.persistence.Transient;


@Entity
public class College {


 private String name;
 private String address;
 private String id;


 public String getId() {
  return id;
 }

 public void setId(String id) {
  this.id = id;
 }

 @Id
 @Column(name="myname")
 public String getName() {
  return name;
 }

 public void setName(String name) {
  this.name = name;
 }
 @Column(name="myadd")
 public String getAddress() {
  return address;
 }
 public void setAddress(String address) {
  this.address = address;
 }
}
```

hibernate will create a table with the name college which contains three columns, related to each field in the above java bean , but if u don't want one field as a column in that table then u can make or associate @Transient annotation on that field as below

```
@Transient
public String getId() {
  return id;
 }
```

### 59. .what is all the inputs we are giving to SessionFactory?

As an input to the Session Factory we are giving hibernate.cfg.xml which contains

- Database connection properties
- Hibernate Configuration
- Mapping Resources(hbm.xml details)

### 60. What we are writing in hibernate-mapping file?

In Hibernate mapping file we need to specify all the properties of our POJO class which object we want to persist.so that based on property it will create column name and table in DB.

Example:

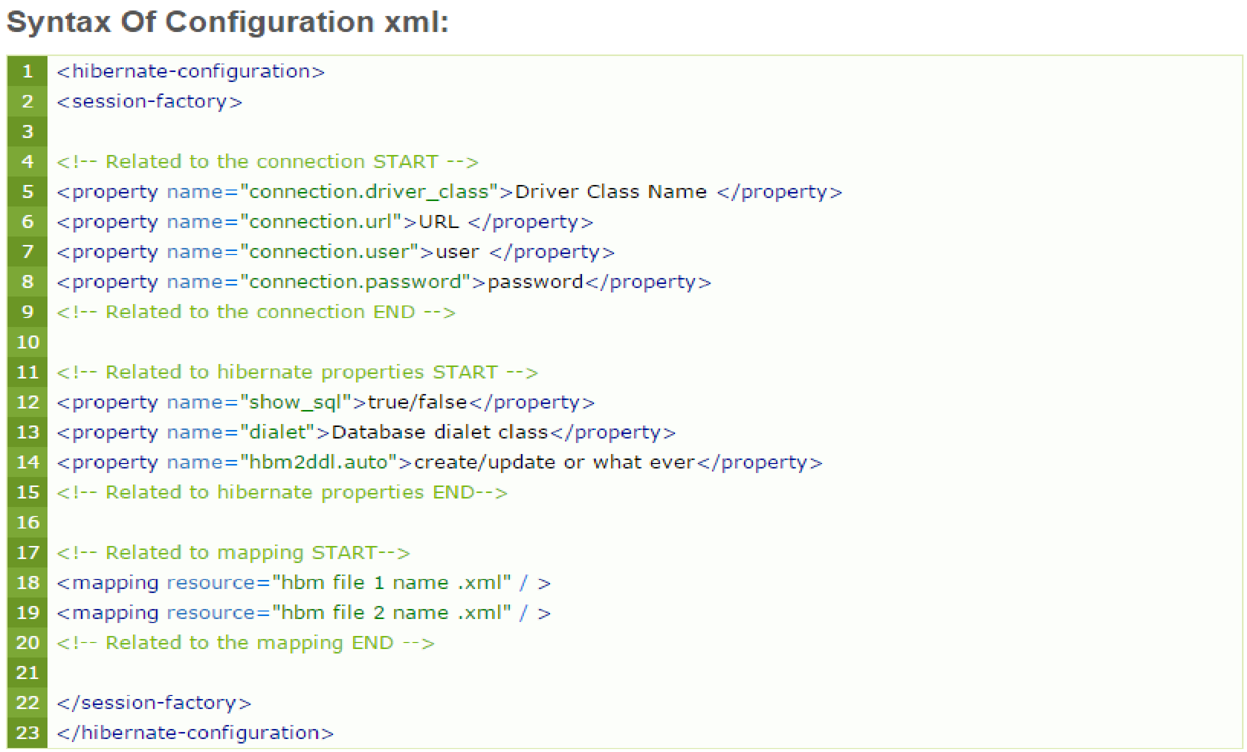
```
public class Employee {
   private int id;
   private String firstName;
   private String lastName;
   private int salary;


   public Employee() {}
```

employee.hbm.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
 "-//Hibernate/Hibernate Mapping DTD//EN"
 "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
   <class name="Employee" table="EMPLOYEE">
      <id name="id" type="int" column="id"/>
      <property name="firstName" column="first_name" type="string"/>
      <property name="lastName" column="last_name" type="string"/>
      <property name="salary" column="salary" type="int"/>
   </class>
</hibernate-mapping>
```

### 61. What we are writing in hibernate-configuration file?

In hibernate-configuration file we need to specify all database connection properties along with hibernate properties and mapping resource/ annotated class as below.



Syntax Of Configuration xml:

```
<hibernate-configuration>
<session-factory>

<!-- Related to the connection START -->
<property name="connection.driver_class">Driver Class Name </property>
<property name="connection.url">URL </property>
<property name="connection.user">user </property>
<property name="connection.password">password</property>
<!-- Related to the connection END -->

<!-- Related to hibernate properties START -->
<property name="show_sql">true/false</property>
<property name="dialet">Database dialet class</property>
<property name="hbm2ddl.auto">create/update or what ever</property>
<!-- Related to hibernate properties END-->

<!-- Related to mapping START-->
<mapping resource="hbm file 1 name .xml" / >
<mapping resource="hbm file 2 name .xml" / >
<!-- Related to the mapping END -->

</session-factory>
</hibernate-configuration>
```

## 62. What is use of @ComponentScan?

Normally @ComponetScan name itself indicate the role of this annotation, it will scan your component class. Simply if we use both annotation approach and java base configuration approach then we have to use this annotation

So if we are using annotation approach in application-context.xml we are writing <context:componetsacn base package=""/> R8, so in java base configuration approach we are not consider any kind of xml then how I have to intimate to IOC that we are using java base configuration along with annotation please kindly scan my annotation bean also .so to overcome this problem spring introduced @Componetscan annotation.

## 63. What is use of Dispatcher servlet?

Dispatcher servlet act as a front controller and single entry way who takes the responsibility to handle the request, normally in Model view Controller architecture always request hits to Dispatcher Servlet then only request will forwarded to next web component. Dispatcher Servlet is a class which provided in org.springframewok.web.servlet.DispatcherServlet package to provide runtime support to my web component

## 64. How to validate valid username and password in spring? For validating can I directly interact with DAO without service?

To validate valid username and password we need to use server side validation where we have to take one more validation Layer we have to take on Customized class implements from Validator (I) provided by spring framework, where we have to Read the data directly from User interface and by implementing Regular Expression we have to write the logic for validation.

## 65. By default servlet container will handle multi-threaded applications, then why you are implementing multi-threading in your application?

Yes By default servlet container provided Multithreading environment but if application access by huge user so incoming request will more which can't handle by Servlet container that's why we have to deploy our code in Cluster environment to balance the load so there we have to implements multithreading approach, Or in case of web services also we need to manually apply Multithreading mechanism to archive asynchronous request/response.