Questions -

Q1) What are the Spring Boot features discussed till now?
Ans: 1. Spring Boot Starters
2. Auto Configuration
3. Centralizing Configuration Data with Property File /YAML

2) What is use of Spring Boot Starters?
Ans:
Each Starter brings some set of Jars and it's Dependent and Compatible jars.

3) What is the Auto Configuration?
Ans:
Beans will be Configured by the Spring Boot automatically without explicit Configuration.
So Bean Instances will be available in Spring Container Readily.
You have to Inject those beans where ever you want.

4) Can I ask the Spring Boot to Auto-Configure CustomerDAOImpl?
Ans: No

5) Is it required to me to Configure CustomerDAOImpl in Bean Configuration class?
Ans: No

6) How to Configure the Beans which I am writing In My applications?
Ans:

A) Some Beans will be Instanciated with ComponentScan facility.
Your Application speciffic beans like Controllers,Business Services,DAO etc can be marked with Steroype annotations called @Controller,@Service, @Repository .

B) Some Beans will be Auto-Configured by the Spring Boot.
All the Built-In Beans like JdbcTemplate, NamedParameterJdbcTemplate,
DataSourceTranactionManager
EntityManagerFactory, EntityManager,
JpaTranactionManager, JpaTemplate
DispatcherServlet, RabbitMqTemplate, KafkaTemplate

C) Some Beans has to be configured by You explicitly.
If above two cases are not possible then you have to configure the Bean explicitly.

Q7) How Spring Boot will decide which bean to Auto-Configured?
Ans:
List of Beans will be Auto-Configured based on the Availability of Bean class in the Class Path.

- Already Some List of Beans are Prepared by Boot Team. (consider 100 Beans)
- At Boot Application Start-Up, Boot Scans All the Packages in the Classpath to Check whether these Bean Classes are available in class path or not.
- Prepares CONDITIONS EVALUATION REPORT with Positive Matches and Negative Maches.
- You can see the CONDITIONS EVALUATION REPORT in debug mode.

Q8) What is the use of Property File / YAML File?
Ans : To Centalize the Bean Configuration Data.

Q9) Can I have both files (properties and yaml)?
Ans: Any one is enough, but you can write both.

Q10) I want to use custom jar in my Spring/Spring Boot application?
Ans:

- Push Your jar into Maven Local repo by specifying group Id and Artifact Id.
- Get that into your project by specifying the group Id and Artifact Id.

Q11) I want to Run My Labs in Oracle. What to do?
Ans: Using Maven Dependency , You can not get the Oracle Jar from Maven Global Repo.

- Download Oracle Jar file.
- Push Your jar into Maven Local repo by Specifing Group Id, Artifact Id,Version
- Specify the Maven Dependency in pom.xml with Above Specified Group Id, Artifact Id,Version
- You can See Oracle Jar in your Project Dependencies.

Q12) What is the Default DataSource in Spring Boot 1.x?
Ans : TomcatCP

Q13) What is the Default DataSource in Spring Boot 2x?
Ans : HikariCP

Q14) Can I Change the Default DataSource to other Required DataSource?
Ans : Yes , You Can . (Refer Lab3 and Lab4)

Q15) What happens when I provide Following 3 DataSources?
HikariCP TomcatCP DBCP2
Ans: HikariCP – Will be Selected **(Refer Lab5)**

Q16) What happens when I provide Following 2 DataSources?
DBCP2 TomcatCP
Ans: TomcatCP – Will be Selected

Q17) What happens when I am not providing Any DataSource after excluding Hikari?
Ans:

Lab3 : Spring Boot and JDBC

Lab3 : Working Steps:

1) Copy Lab2 and Paste as Lab3
2) Update the pom.xml
   A) Exclude HikariCP

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
<exclusions>
<exclusion>
<groupId>com.zaxxer</groupId>
<artifactId>HikariCP</artifactId>
</exclusion>
</exclusions>
</dependency>
```

   B) Add Dependency for Tomcat CP.

```
<dependency>
<groupId>org.apache.tomcat</groupId>
<artifactId>tomcat-jdbc</artifactId>
</dependency>
```

3) Update **application.properties**
   A) Remove Hikari Properties
   B) Add Tomcat Properties as follows

```
spring.datasource.tomcat.initialSize=5
spring.datasource.tomcat.max-active=25
```

4) Run the Application.
   Run As => Spring Boot Application

Lab3 : Spring Boot–JDBC Example

Lab3: Files required

1. **MyBootApp.java**	**Same As Lab2**
2. **CustomerDAO.java**	**Same As Lab2**
3. **CustomerDAOImpl.java**	**Same As Lab2**
4. **Customer.java**	**Same As Lab2**
5. **CustomerRowMapper.java**	**Same As Lab2**
6. **application.properties**	**Updated**
7. **pom.xml**	**Updated**

6) application.properties

#Database Properties
spring.datasource.url=jdbc:mysql://localhost:3306/myjtcdb
spring.datasource.username=root
spring.datasource.password=SomPrakash

#Tomcat Properties
spring.datasource.tomcat.initialSize=5
spring.datasource.tomcat.max-active=25

#Log Properties
logging.level.root=INFO
logging.pattern.console=%-5level %logger{36} - %msg%n

7) pom.xml

<project .....>
<modelVersion>4.0.0</modelVersion>

<groupId>com.coursecube.springboot</groupId>
<artifactId>**Lab3**</artifactId>
<version>1.0</version>
<packaging>jar</packaging>
<name>**Lab3**</name>
<url>http://www.jtcindia.org</url>

<parent>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-parent</artifactId>
<version>2.1.2.RELEASE</version>
</parent>

```
<dependencies>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
<exclusions>
<exclusion>
<groupId>com.zaxxer</groupId>
<artifactId>HikariCP</artifactId>
</exclusion>
</exclusions>
</dependency>

<dependency>
<groupId>org.apache.tomcat</groupId>
<artifactId>tomcat-jdbc</artifactId>
</dependency>

<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>5.1.6</version>
</dependency>

</dependencies>
<build>
<plugins>
<plugin>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>
</project>
```

Lab4 : Working Steps:

1) Copy Lab2 and Paste as Lab4
2) Update the pom.xml

A) Exclude HikariCP

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
<exclusions>
<exclusion>
<groupId>com.zaxxer</groupId>
<artifactId>HikariCP</artifactId>
</exclusion>
</exclusions>
</dependency>
```

B) **Add Dependency for DBCP2**

```
<dependency>
<groupId>org.apache.commons</groupId>
<artifactId>commons-dbcp2</artifactId>
</dependency>
```

3) Update **application.properties**

C) Remove Tomcat Properties

D) Add DBCP2 Properties as follows

```
spring.datasource.dbcp2.initial-size=5
spring.datasource.dbcp2.max-total=20
spring.datasource.dbcp2.pool-prepared-statements=true
```

4) Run the Application.

Run As => Spring Boot Application

Lab4: Files required

1. **MyBootApp.java**	**Same As Lab2**
2. **CustomerDAO.java**	**Same As Lab2**
3. **CustomerDAOImpl.java**	**Same As Lab2**
4. **Customer.java**	**Same As Lab2**
5. **CustomerRowMapper.java**	**Same As Lab2**
6. **application.properties**	**Updated**
7. **pom.xml**	**Updated**

6) application.properties

#Database Properties
spring.datasource.url=jdbc:mysql://localhost:3306/myjtcdb
spring.datasource.username=root
spring.datasource.password=SomPrakash

#DBCP2 Properties
spring.datasource.dbcp2.initial-size=5
spring.datasource.dbcp2.max-total=20
spring.datasource.dbcp2.pool-prepared-statements=true

#Log Properties
logging.level.root=INFO
logging.pattern.console=%-5level %logger{36} - %msg%n

7) pom.xml

<project .....>
<modelVersion>4.0.0</modelVersion>
<groupId>com.coursecube.springboot</groupId>
<artifactId>**Lab4**</artifactId>
<version>1.0</version>
<packaging>jar</packaging>
<name>**Lab4**</name>
<url>http://www.jtcindia.org</url>
<parent>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-parent</artifactId>
<version>2.1.2.RELEASE</version>
</parent>

```
<dependencies>

<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
<exclusions>
<exclusion>
<groupId>com.zaxxer</groupId>
<artifactId>HikariCP</artifactId>
</exclusion>
</exclusions>
</dependency>

<dependency>
<groupId>org.apache.commons</groupId>
<artifactId>commons-dbcp2</artifactId>
</dependency>

<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>5.1.6</version>
</dependency>

</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>
</build>
</project>
```

Lab5 : Working Steps:

1) Copy Lab2 and Paste as Lab5
2) Update the pom.xml

A) Include HikariCP

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
</dependency>
```

B) Add Dependency for DBCP2

```
<dependency>
<groupId>org.apache.commons</groupId>
<artifactId>commons-dbcp2</artifactId>
</dependency>
```

C) Add Dependency for Tomcat CP.

```
<dependency>
<groupId>org.apache.tomcat</groupId>
<artifactId>tomcat-jdbc</artifactId>
</dependency>
```

3) Update **application.properties**

A) Add HIkari Properties as follows

```
spring.datasource.hikari.connectionTimeout=20000
spring.datasource.hikari.maximumPoolSize=5
```

B) Add Tomcat Properties as follows

```
spring.datasource.tomcat.initialSize=5
spring.datasource.tomcat.max-active=25
```

C) Add DBCP2 Properties as follows

```
spring.datasource.dbcp2.initial-size=5
spring.datasource.dbcp2.max-total=20
spring.datasource.dbcp2.pool-prepared-statements=true
```

4) Run the Application.

Run As => Spring Boot Application

Lab5: Files required

1. MyBootApp.java	Same As Lab2
2. CustomerDAO.java	Same As Lab2
3. CustomerDAOImpl.java	Same As Lab2
4. Customer.java	Same As Lab2
5. CustomerRowMapper.java	Same As Lab2
6. application.properties	Updated
7. pom.xml	Updated

8) application.properties

#Database Properties
spring.datasource.url=jdbc:mysql://localhost:3306/myjtcdb
spring.datasource.username=root
spring.datasource.password=SomPrakash

#Hikari Properties
spring.datasource.hikari.connectionTimeout=20000
spring.datasource.hikari.maximumPoolSize=5

#Tomcat Properties
spring.datasource.tomcat.initialSize=5
spring.datasource.tomcat.max-active=25

#DBCP2 Properties
spring.datasource.dbcp2.initial-size=5
spring.datasource.dbcp2.max-total=20
spring.datasource.dbcp2.pool-prepared-statements=true

#Log Properties
logging.level.root=INFO
logging.pattern.console=%-5level %logger{36} - %msg%n

9) pom.xml

<project .....>
<modelVersion>4.0.0</modelVersion>
<groupId>com.coursecube.springboot</groupId>
<artifactId>**Lab5**</artifactId>
<version>1.0</version>
<packaging>jar</packaging>

```
<name>Lab5</name>
<url>http://www.jtcindia.org</url>
<parent>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-parent</artifactId>
<version>2.1.2.RELEASE</version>
</parent>

<dependencies>

<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-jdbc</artifactId>
</dependency>

<dependency>
<groupId>org.apache.tomcat</groupId>
<artifactId>tomcat-jdbc</artifactId>
</dependency>

<dependency>
<groupId>org.apache.commons</groupId>
<artifactId>commons-dbcp2</artifactId>
</dependency>

<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>5.1.6</version>
</dependency>

</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>
</build>
</project>
```