1. Create a Restful API using Spring Boot for Student.(1 Mark)

[{"rollNumber":1,"name":"Mehak","classAndSection":"12 A"},{"rollNumber":4,"name":"Sagar","classAndSection":"11 A"},
{"rollNumber":45,"name":"Shubham","classAndSection":"9 C"},{"rollNumber":25,"name":"Gaurav","classAndSection":"11 D"},
{"rollNumber":19,"name":"Keshav","classAndSection":"10 B"},{"rollNumber":67,"name":"Bhavna","classAndSection":"11 A"}]
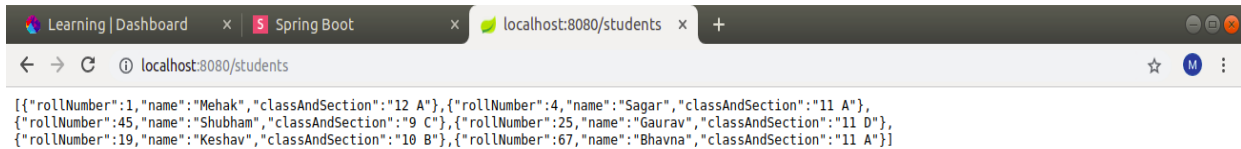
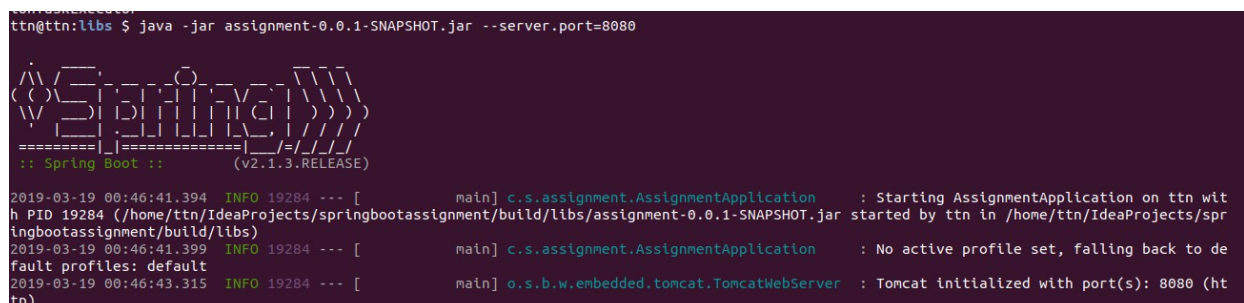2. Run Spring Boot Application with all the three ways (1 Mark)

• Run the Main java class marked with @SpringBootApplication Annotation and contains main method as follows:

• Execute the bootRun Gradle tasks:

• Create an Executable jar with the help of bootJar Gradle task. Now execute the jar files using command below:

3. Create Bean User containing two field username and password with Spring Context File (1 Mark)

```xml
1    <?xml version="1.0" encoding="UTF-8"?>
2    <beans xmlns="http://www.springframework.org/schema/beans"
3           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4           xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-bean
5
6        <bean class="com.springboot.assignment.Question3.User" id="user">
7            <property name="username" value="adlakhamehak98"/>
8            <property name="password" value="12345@ttn"/>
9        </bean>
10
11   </beans>
```

4. Create a Bean Database with two fields port and name and Access its values using application.properties (1 Mark)

```java
@Component
public class Database {

    @Value("${app.name}")
    String name;

    @Value("${app.port}")
    String port;

    public String getName() {
        return name;
    }

    public String getPort() {
        return port;
    }
}
```
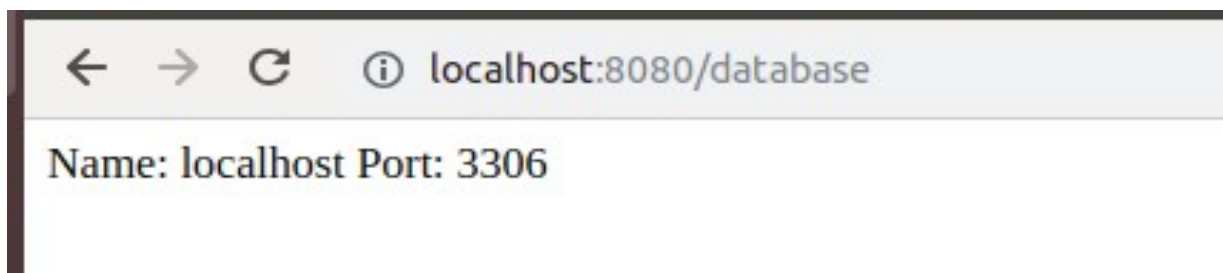
← → C ⓘ localhost:8080/database

Name: localhost Port: 3306

5. Configure environment specfic values form Database Bean (2 Marks)

```
ttn@ttn:libs $ java -jar assignment-0.0.1-SNAPSHOT.jar --spring.profiles.active=dev

 /\\ /___'_ _ _ _()_ _ _ _ _(_)_ _ _ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v2.1.3.RELEASE)
```

6. Apply Logging whereever you feel is necessity (1 Mark)

```
2019-03-21 20:34:59.876  INFO 12832 --- [nio-8080-exec-1] c.s.assignment.Service.DatabaseService   : getting database name and port from DatabaseService
2019-03-21 20:35:05.917  INFO 12832 --- [nio-8080-exec-2] c.s.assignment.Service.StudentService    : adding data in Student through StudentService
2019-03-21 20:35:12.327  INFO 12832 --- [nio-8080-exec-4] c.s.assignment.Service.StudentService    : getting bootstrapped data in Student from StudentService
```

7. Bootstrap Data for Student Domain (3 Marks)

```java
//Bootstrap Data for Student Domain (3 Marks)
@Component
public class StudentBootstrap {
    @Autowired
    StudentRepository studentRepository;

    @EventListener(ApplicationStartedEvent.class)
    public void init() {
        Iterator<Student> iterable = studentRepository.findAll().iterator();
        if (!iterable.hasNext()) {
            for (int i = 1; i <= 10; i++) {
                Student student = new Student(i, name: "Name "+i, classAndSection: "Class "+i);
                studentRepository.save(student);
            }
        }
        System.out.println("Your Application is up and running");
    }
}
```