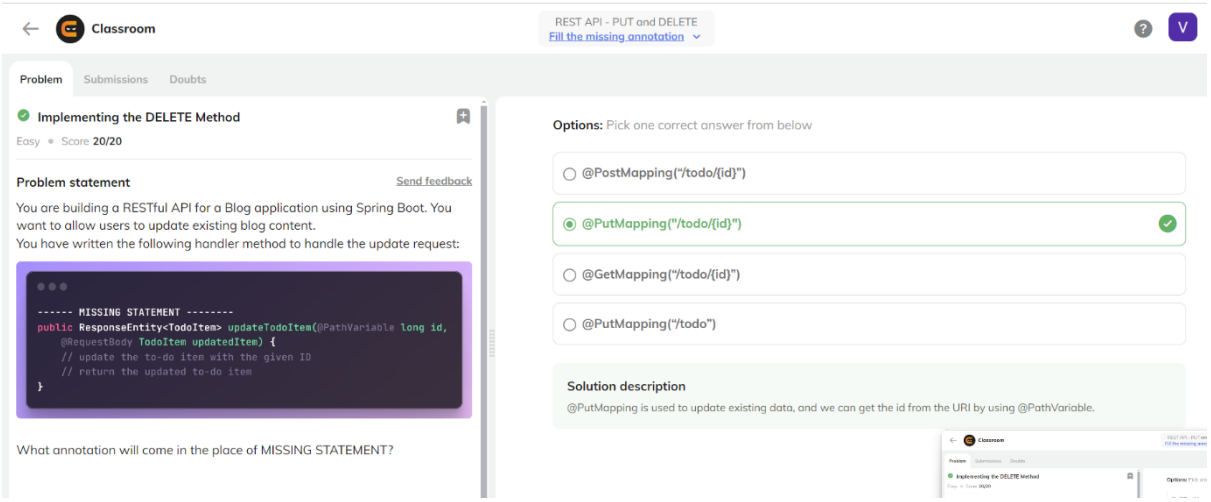
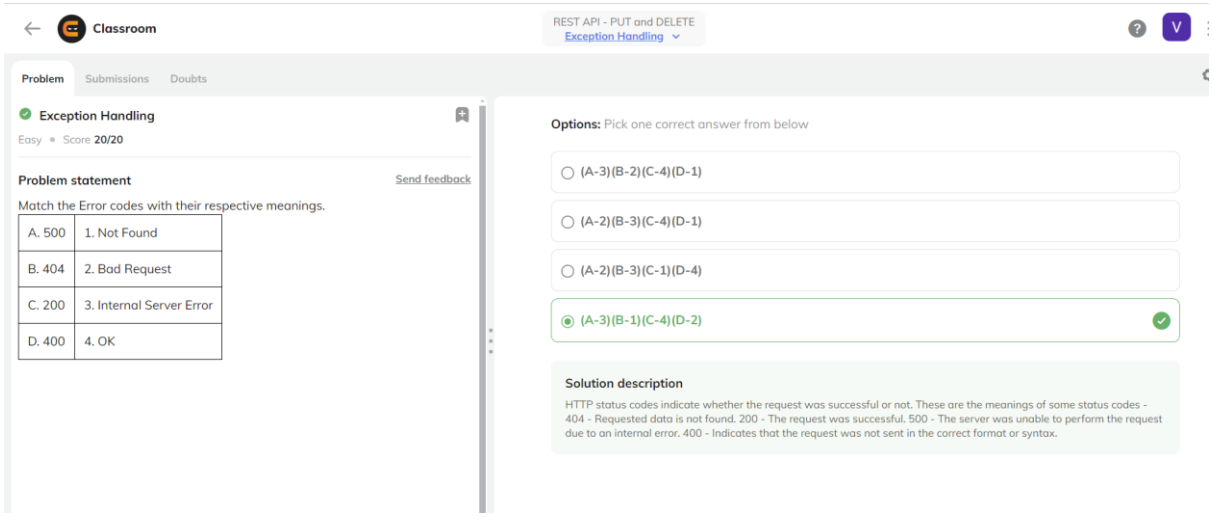
Quiz Questions

(1) Implementing the Delete Method



(2) Exception Handling



(3) Exception Handling

A.

```
if(ObjectUtils.isEmpty(blogMap.get(id))) {
    // throw Exception
}
```

B.

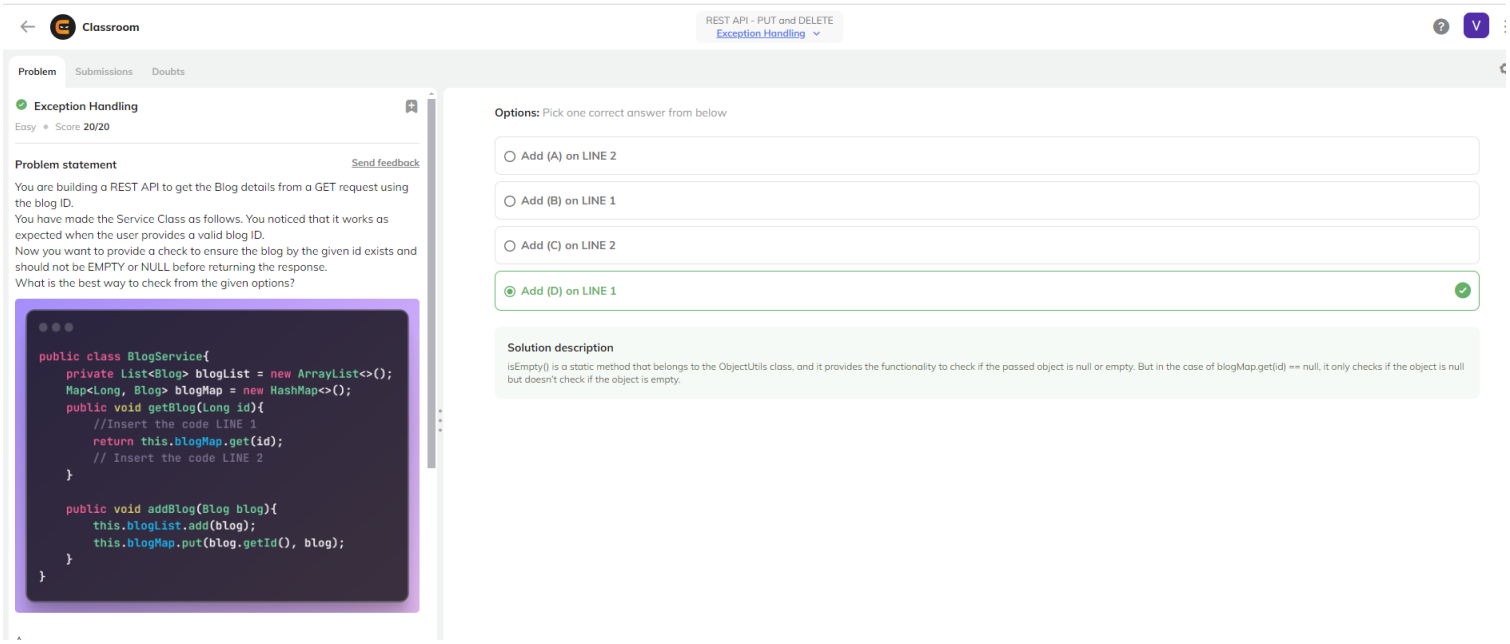
```
if(blogMap.get(id) == null){
    // throw Exception
}
```

C.

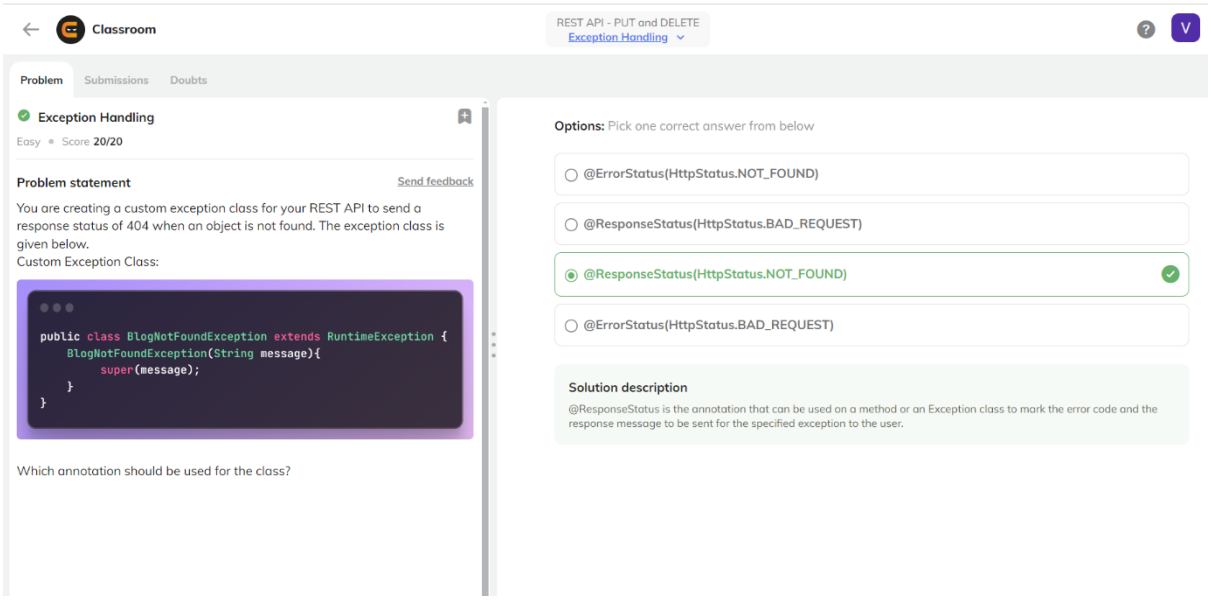
```
if(blogMap.get(id) == null){
    // throw Exception
}
```

D.

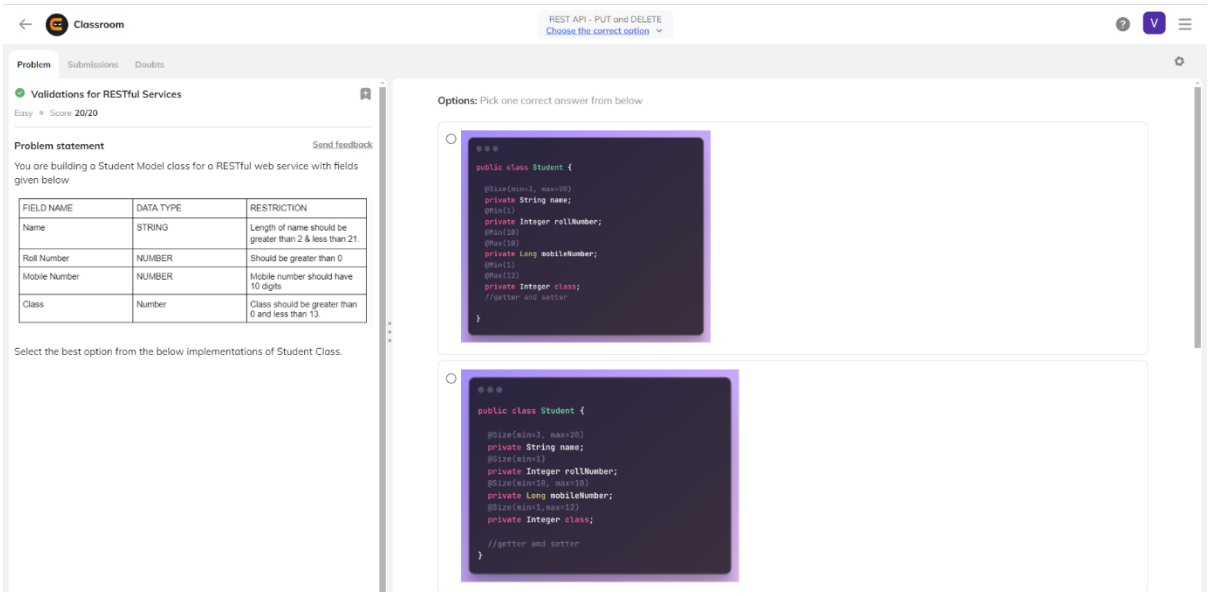
```
if(ObjectUtils.isEmpty(blogMap.get(id))) {
    // throw Exception
}
```



(4) Exception Handling



(5) Validation for RESTful Services




```
public class Student {

  @Size(min=3, max=20)
  private String name;
  @Min(1)
  private Integer rollNumber;
  @Min(1000000000)
  @Max(9999999999)
  private Long mobileNumber;
  @Min(1)
  @Max(12)
  private Integer class;

  //getter and setter

}
```

```
public class Student {

  @Min(3)
  @Max(20)
  private String name;
  @Min(1)
  private Integer rollNumber;
  @Size(min=10, max=10)
  private Long mobileNumber;
  @Min(1)
  @Max(12)
  private Integer class;

  //getter and setter

}
```

Solution description

@Size annotation is used to constrain the length of the field, and the @Min is used on a number (byte, short, long, int and their respective wrappers) to restrict their values greater than or equal to the specified value. Similarly, @Max is used to restrict the number value should be less than or equal to the specified value.