

```
driver - Notepad
File Edit Format View Help
package com.spring.main;

import java.util.Scanner;
import org.springframework.context.ApplicationContext;
import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class Driver {
    public static void main(String[] args) {
        // fill the code
        Scanner sc=new Scanner(System.in);
        ApplicationContext context= new AnnotationConfigApplicationContext(ApplicationConfig.class);

        System.out.println("Enter the student name:");
        String name=sc.nextLine();
        System.out.println("Enter the admission number:");
        String admissionNo=sc.nextLine();
        System.out.println("Enter the grade:");
        char grade= sc.nextChar();

        System.out.println("Enter the hostel details");
        System.out.println("Enter the hostel name:");
        String hostelName=sc.nextLine();
        System.out.println("Enter the room rent:");
        float rent=sc.nextFloat();

        StudentService ss=new StudentService();
        ss.calculateFee (name, admissionNo, grade, hostelName, rent);
    }
}
```

```
studentBO - Notepad
File Edit Format View Help
package com.spring.bo;

import com.spring.model.Student;

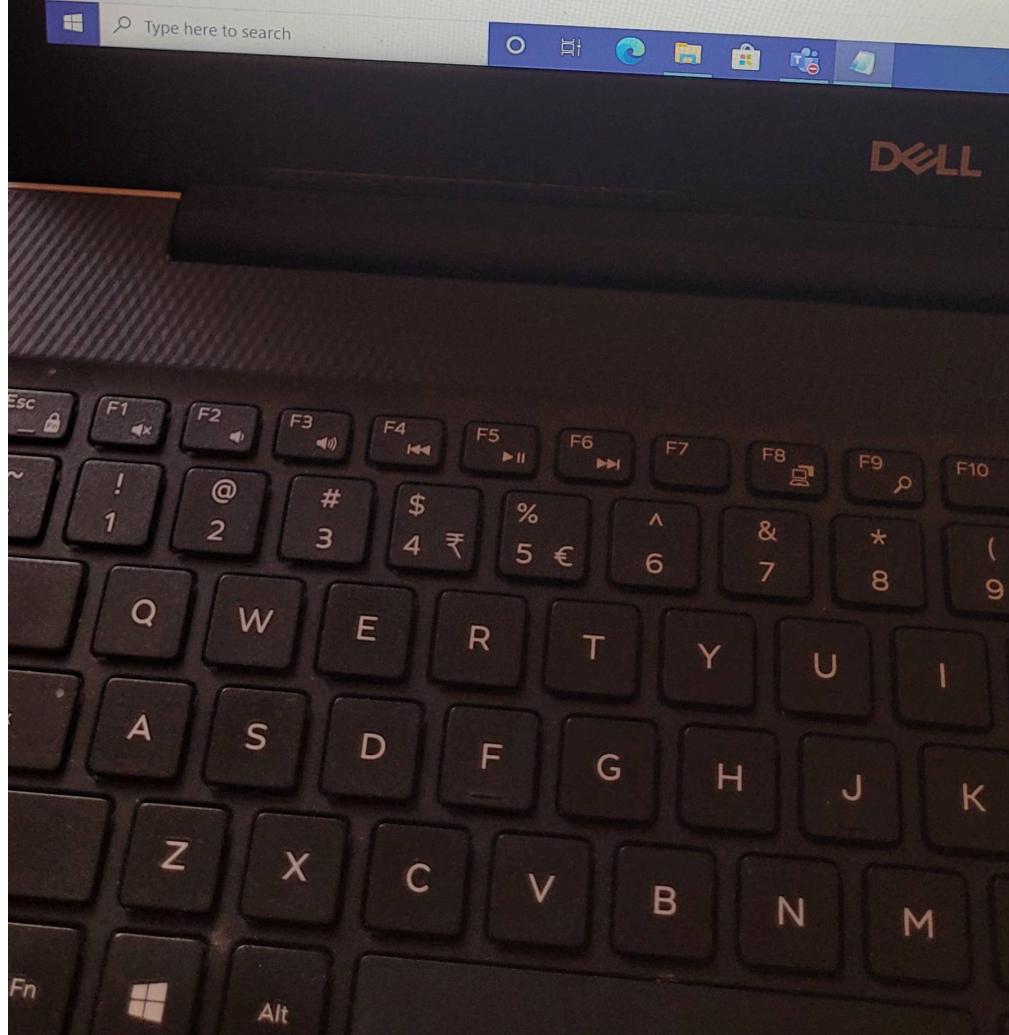
import org.springframework.stereotype.Component;
//use appropriate annotation to make this class as component class
@Component
public class StudentBO {

    public float calculateFee(Student obj) {
        float fee=0;
        //fill the code

        if(Student.getGrade()=='A'){
            fee = (Student.getHostel.getRent()-(Student.getHostel.getRent()*30/100));
        }else if(Student.getGrade()=='O'){
            fee= (Student.getHostel.getRent()-(Student.getHostel.getRent()*50/100));
        }else if(Student.getGrade()=='D'){
            fee=(Student.getHostel.getRent()-(Student.getHostel.getRent()*40/100));
        }else if(Student.getGrade()=='B'){
            fee=(Student.getHostel.getRent()-(Student.getHostel.getRent()*20/100));
        }else if(Student.getGrade()=='C' || Student.getGrade()=='F'){
            fee=Student.getHostel.getRent();
        }

        return fee;
    }

}
```



StudentService - Notepad

File Edit Format View Help

```
package com.spring.service;
```

```
import com.spring.exception.InvalidGradeException;  
  
import com.spring.bo.StudentBO ;  
import org.springframework.stereotype.Component;  
import org.springframework.beans.factory.annotation.Autowired;  
  
//use appropriate annotation to make this class as component class  
@Component  
public class StudentService {  
  
    public StudentBO getStudentBOObj() {  
        return studentBOObj;  
    }  
  
    public void setStudentBOObj(StudentBO studentBOObj) {  
        this.studentBOObj = studentBOObj;  
    }  
  
    private StudentBO studentBOObj;  
  
    //fill the code  
    @Autowired  
    public StudentService(StudentBO studentBOObj) {  
        super();  
        this.studentBOObj = studentBOObj;  
    }  
  
    public float calculateFee(String name, String admissionNo, char grade, String hostelName, float rent) {  
        //fill the code  
  
        if(grade=='O' || grade=='A' || grade=='D' || grade=='B' || grade=='C' || grade=='F'){  
            Student st=new Student();  
            st.setName(name);  
            st.setAdmissionNo(admissionNo);  
            st.setGrade(grade);  
            st.setHostelName(hostelName);  
            st.setRent(rent);  
  
            studentBOObj.calculateFee(st);  
        }  
        else{  
            throws new InvalidGradeException("Invalid Grade");  
        }  
  
        return 0;  
    }  
}
```



Type here to search



& Run Evaluate Full screen Description Assessment Home

FeesDiscountData... Hostel.java Student.java StudentService.java

```
1 package com.spring.model;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4 import org.springframework.stereotype.Component;
5 //pojo class with required attributes, getters and setters
6 //use appropriate annotation to make this class as component class
7 @Component
8 public class Student {
9
10     private String name;
11     private String admissionNo;
12     private char grade;
13     private Hostel hostel;
14     //fill the code
15     @Autowired
16     public Student(Hostel hostel) {
17         super();
18         this.hostel = hostel;
19     }
20     public Hostel getHostel() {
21         return hostel;
22     }
23     public void setHostel(Hostel hostel) {
24         this.hostel = hostel;
25     }
}
```

Note: Ensure to provide all your program's inputs in the input field.

the Student object and set the name, admissionNo, hostelName, rent in that object. In case the grade is user-defined Exception InvalidGradeException should be thrown with message "Invalid Grade".

If the validation is done call the method calculateFee in StudentBO class and perform the implementation.

Please editor before Compile & Run.
Assessment.

Limitations and Constraints:**Full screen****Description**

Driver.java

FeesDiscountData...

Hostel.java

Student.java

StudentSer...

```
1 package com.spring.model;
2
3 import org.springframework.stereotype.Component;
4 //pojo class with required attributes, getters and setters
5 //use appropriate annotation to make this class as component class
6 @Component
7 public class Hostel {
8
9     private String hostelName;
10    private float rent;
11
12    public String getHostelName() {
13        return hostelName;
14    }
15    public void setHostelName(String hostelName) {
16        this.hostelName = hostelName;
17    }
18    public float getRent() {
19        return rent;
20    }
21    public void setRent(float rent) {
22        this.rent = rent;
23    }
24 }
```

53123

Note: Ensure to provide all your program's inputs at a single stretch before clicking "Compile and Run".
Eg: Program expects an input of ID, Name and Phone which displays all these details in the console as output, then the inputs must be given as:



DELL

d=53123#

Cognizant |

the Student object and set the name, admission hostelName, rent in that object. In case the grade user-defined Exception InvalidGradeException is thrown with message "Invalid Grade".

If the validation is done call the method calculateFees in StudentBO class and perform the implementation.

Limitations and Constraints:

code editor before Compile & Run.
ie assessment.

te

Full screen

Description

InvalidGradeExce... ▾ Driver.java ▾ FeesDiscountData... ▾ Hostel.java ▾ Studen...

```
1 package com.spring.model;
2
3 import java.util.Map;
4
5 import org.springframework.stereotype.Component;
6 //pojo class with required attributes, getters and setters
7 //use appropriate annotation to make this class as component class
8 @Component
9 public class FeesDiscountDetails {
10
11
12     //fill the code
13     @Value("#{${feesdiscount.map}}")
14     private Map<String, Integer> feesDiscount;
15
16     public Map<String, Integer> getfeesDiscount() {
17         return feesDiscount;
18     }
19
20     public void setFeesDiscount(Map<String, Integer> feesDiscount) {
21         this.feesDiscount = feesDiscount;
22     }
23
24 }
```

Note: Ensure to provide all your program's inputs at a single stretch before clicking "Compile and Run".
Eg: Program expects an input of ID Name and Phone, which displays all these data in the console as output, then the inputs must be given as

DELL

Question Description

Getter and setter method for the above attribute part of code skeleton. One argument constructor part of code skeleton. The **StudentBO** object be autowired above the constructor via annotation.

Spring Framework / Hostel Management V1

Please edit before Compile & Run.
assessment.

In this **StudentService** class, the method **public void calculateFee (String name, String admissionGrade, String hostelName, float rent)** accepts name, admissionGrade, hostelName, rent as the arguments .Validation grade values should be 'O', 'D', 'A', 'B', 'C', 'F' if the Student object and set the name, admissionGrade, hostelName, rent in that object. In case the grade is user-defined Exception **InvalidGradeException** is thrown with message "Invalid Grade".

If the validation is done call the method **calculateFee** in **StudentBO** class and perform the implementation.

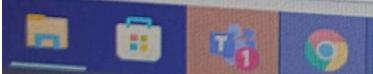
Limitations and Constraints:

Full screen

Description

StudentBO.java ApplicationConfig... InvalidGradeException.java ★ Driver.java Fees

```
1 package com.spring.exception;
2
3 public class InvalidGradeException extends Exception {
4     public InvalidGradeException(string msg) {
5         // fill the code
6         super(msg);
7     }
8 }
```



DELL

j=53123#

Cognizant |

Spring Framework / Hostel Management V1

code editor before Compile & Run.
he assessment.

Full screen **Description**

pom.xml StudentBO.java ApplicationConfig... InvalidGradeExce... feesDiscountData...

```
1 package com.spring.config;
2
3 import org.springframework.context.annotation.ComponentScan;
4 import org.springframework.context.annotation.Configuration;
5 //Use appropriate annotation
6
7 @Configuration
8 @ComponentScan
9 public class ApplicationConfig {
10
11
12 }
13
```

Question Description

Getter and setter method for the above attribute part of code skeleton. One argument constructor part of code skeleton. The **StudentBO** object should be **autowired** above the **constructor** via annotation.

In this **StudentService** class, the method **public float calculateFee (String name, String admissionNo, String hostelName, float rent)** accepts name, admissionNo, grade, hostelName, rent as the arguments. Validate the grade values should be 'O', 'D', 'A', 'B', 'C', 'F' if the grade is invalid then create a new **Student** object and set the name, admissionNo, hostelName, rent in that object. In case the grade is invalid then throw user-defined Exception **InvalidGradeException** should be thrown with message "**Invalid Grade**".

If the validation is done call the method **calculateFee()** in **StudentBO** class and perform the implementation.

Limitations and Constraints:

DELL