

NCC Cadet Details(---RETIRED---)

Grade settings: Maximum grade: 100

Disable external file upload, paste and drop external content: Yes

Based on: [NCC Cadet Details\(---RETIRED---\)](#)

Run: Yes **Evaluate:** Yes

Automatic grade: Yes

The NCC organisation desired to calculate Cadet's fees based on Cadet's age. The NCC organization's manager contacts a software developer for assistance with their process. You are the software developer, and you are developing a Java program in accordance with the specifications

Component Specification: Cadet

Type (Class)	Attributes	Methods
Cadet	String cadetNumber String cadetName String groupName String gender int age	Necessary getters, setters, and a five-argument constructor is provided as a part of the code skeleton.

Functional Requirement 1: Extract the details of the Cadet and create an object for Cadet class.

Type (Class)	Methods	Responsibilities
UserInterface	public static Cadet extractDetails (String cadetDetails)	This method accepts cadetDetails separated by the colon as an argument and should extract the properties of the Cadet from the argument by parsing the cadetDetails. Set

		these values to the Cadet object and return this object.
--	--	--

Functional Requirement 2: Calculate the Cadet's fees based on their age.

Type (Class)	Methods	Responsibilities
Cadet	public double calculateFeesToBePaid()	<p>This method is used to calculate the Cadets' fees based on their age.</p> <p>If the age is between 11 and 15, the fee amount is 2500.</p> <p>If the age is between 16 and 20, the fee amount is 3000.</p> <p>If the age is between 21 and 25, the fee amount is 3200.</p> <p>If the age is between 26 and 30, the fee amount is 3800.</p> <p>If the age is between 31 and 35, the fee amount is 4200.</p> <p>Condition:</p> <ul style="list-style-type: none"> • If the age does not belong to any of the above-mentioned ranges, then return -1. • Both the ages mentioned in the range are inclusive. • The groupName should be "Trollers" else return -1. • groupName is case-insensitive.

The main method in the **UserInterface** class is excluded from the evaluation. You are free to write your own code in the main method to invoke the business methods to check its correctness.

Note:

- In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the rest of the text represents the output.
- Ensure to follow the object-oriented specifications provided in the question.
- Ensure to provide the names for classes, attributes, and methods as specified in the question.
- Adhere to the code template, if provided.

Sample Input / Output 1:

Enter Cadet Details

CADET344:Danny:Trollers:Male:23

Cadet Number : CADET344

Cadet Name : Danny

Group Name : Trollers

Gender : Male

Age : 23

Fees to be paid by the Cadet : 3200.00

Explanation: Since the group name is Trollers and the age is between 21 and 25 fees to be paid is 3200.0

Sample Input / Output 2:

Enter Cadet Details

CADET287:Kenny:Trollers:Male:-9

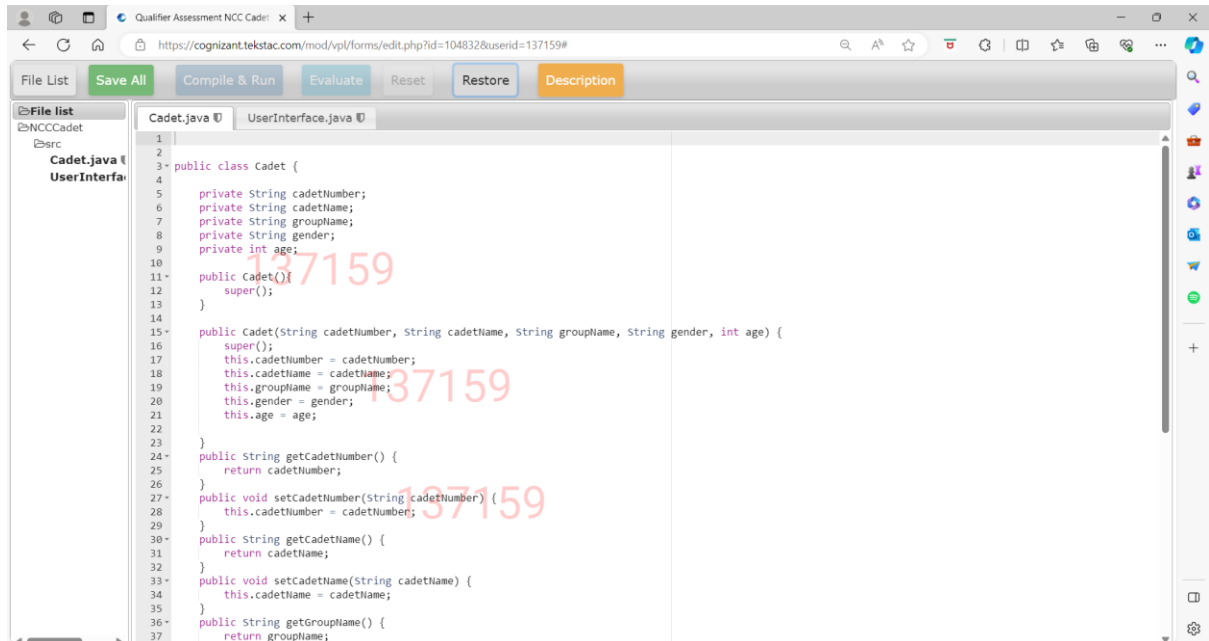
Invalid Cadet Details

Sample Input / Output 3:

Enter Cadet Details

CADET287:Kenny:Rockers:Male:29

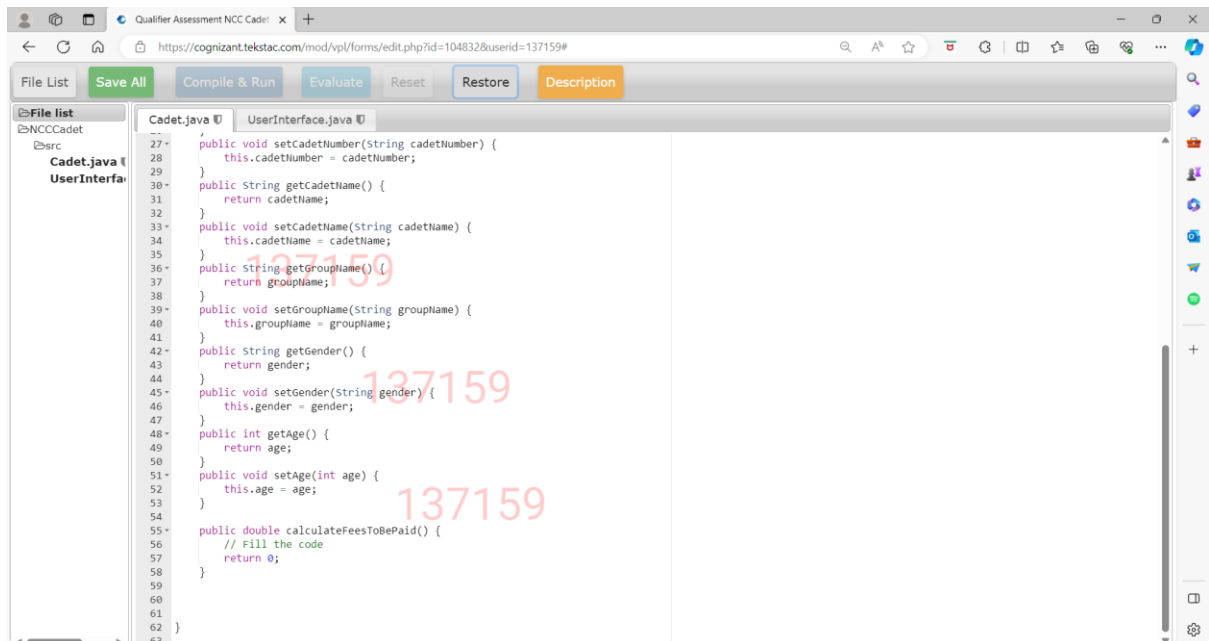
Invalid Cadet Details



The screenshot shows an IDE window with the following elements:

- Browser tab: Qualifier Assessment NCC Cadet
- URL: <https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=104832&userid=137159#>
- Buttons: File List, Save All, Compile & Run, Evaluate, Reset, Restore, Description
- File List: NCCCadet, src, Cadet.java, UserInterface.java
- Code Editor (Cadet.java):

```
1 public class Cadet {
2
3     private String cadetNumber;
4     private String cadetName;
5     private String groupName;
6     private String gender;
7     private int age;
8
9     public Cadet() {
10         super();
11     }
12
13     public Cadet(String cadetNumber, String cadetName, String groupName, String gender, int age) {
14         super();
15         this.cadetNumber = cadetNumber;
16         this.cadetName = cadetName;
17         this.groupName = groupName;
18         this.gender = gender;
19         this.age = age;
20     }
21
22     public String getCadetNumber() {
23         return cadetNumber;
24     }
25
26     public void setCadetNumber(String cadetNumber) {
27         this.cadetNumber = cadetNumber;
28     }
29
30     public String getCadetName() {
31         return cadetName;
32     }
33
34     public void setCadetName(String cadetName) {
35         this.cadetName = cadetName;
36     }
37
38     public String getGroupName() {
39         return groupName;
40     }
41 }
```



The screenshot shows an IDE window with the following elements:

- Browser tab: Qualifier Assessment NCC Cadet
- URL: <https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=104832&userid=137159#>
- Buttons: File List, Save All, Compile & Run, Evaluate, Reset, Restore, Description
- File List: NCCCadet, src, Cadet.java, UserInterface.java
- Code Editor (UserInterface.java):

```
1 public void setCadetNumber(String cadetNumber) {
2     this.cadetNumber = cadetNumber;
3 }
4
5 public String getCadetName() {
6     return cadetName;
7 }
8
9 public void setCadetName(String cadetName) {
10     this.cadetName = cadetName;
11 }
12
13 public String getGroupName() {
14     return groupName;
15 }
16
17 public void setGroupName(String groupName) {
18     this.groupName = groupName;
19 }
20
21 public String getGender() {
22     return gender;
23 }
24
25 public void setGender(String gender) {
26     this.gender = gender;
27 }
28
29 public int getAge() {
30     return age;
31 }
32
33 public void setAge(int age) {
34     this.age = age;
35 }
36
37 public double calculateFeesToBePaid() {
38     // Fill the code
39     return 0;
40 }
41 }
```

