exercise-8

November 26, 2023

0.1 Short Assessed Exercise

0.2 Level 8

0.3 Taher Ahmed

0.4 230288917

0.5 30/12/23

0.6 1

0.7 Summary of the Question

The goal is to develop software that evaluates the accessibility of restaurants for individuals with disabilities. It assesses three aspects: the presence of steps, the availability of toilets, and the availability of disabled parking. Each category is assigned a score from 1 to 3. An "OUTSTANDING" rating is given to restaurants that achieve a score of 9, a "GOOD" rating for scores above 5, and a "POOR" rating for scores of 5 or below. Restaurants without ratings have a score of 0 and are labeled as "UNRATED."

To accomplish this, the software defines a data structure (a class with no methods) to represent restaurant information. This structure includes fields for the restaurant's name, scores in each category, total score, and calculated rating (initially set as UNRATED). The software provides a method called "create" to initialize restaurant records and set their names. Another method allows users to assign ratings based on the three scores by calculating the score and setting the rating. Validations are in place to ensure that scores fall within the range of 1 to 3.

Additionally, a method is provided that converts the record into a formatted string, such as "Posh Nosh has received a disability rating of GOOD." The software can independently display ratings by calling this method.

The program doesn't offer any options to individually modify or access score fields or alter the rating except through the designated method. This ensures that as long as the program relies on the given methods, both the total and rating will always align with the scores consistently.

0.8 Justification that the program passes this level

This is based on tick boxes at start of the question description for this exercise. - I can EXPLAIN HOW MY PROGRAM WORKS - I wrote this program myself Define and use an abstract data type with record fields accessed ONLY by procedural programming style accessor methods including a create method, all defined in the main class. Decomposed fully into multiple methods, call them

with multiple arguments and return values from them All methods individually and helpfully commented on their use (not over-commenting). Good indentation, comments and variable/method naming. All variables have minimal scope and final variables used for literal constants

0.9 The literate program development

0.10 Method name

create_restaurant_record ## What it does A create method should be provided to create initial unrated records and set the restaurant name (passed as an argument to the method). The initial scores should be 0 (meaning it is UNRATED). ## Implementation (how it works) A class called Restaurant is created with relevant attributes and setter and getter (accessor method).

[ ]:
```
class Restaurant {
    private String name;
    private int step_free_access_score;
    private int toilet_score;
    private int parking_score;
    private int total_score;
    private String rating;
    // Accessor method to get the restaurant's name
    public String get_name() {
        return name;
    }

    // Accessor method to get the step_free_access_score
    public int get_step_free_access_score() {
        return step_free_access_score;
    }

    // Accessor method to get the toilet_score
    public int get_toilet_score() {
        return toilet_score;
    }

    // Accessor method to get the parking_score
    public int get_parking_score() {
        return parking_score;
    }

    // Accessor method to get the total_score
    public int get_total_score() {
        return total_score;
    }

    // Accessor method to get the rating
    public String get_rating() {
        return rating;
```

```
    }
    public void set_name(String name) {
        this.name = name;
    }

    public void set_step_free_access_score(int step_free_access_score) {
        this.step_free_access_score = step_free_access_score;
    }

    public void set_toilet_score(int toilet_score) {
        this.toilet_score = toilet_score;
    }

    public void set_parking_score(int parking_score) {
        this.parking_score = parking_score;
    }

    public void set_total_score(int total_score) {
        this.total_score = total_score;
    }

    public void set_rating(String rating) {
        this.rating = rating;
    }
}
public Restaurant create_restaurant_record(String name) {
    Restaurant restaurant = new Restaurant();
    restaurant.set_name(name);
    // set all the scores to zero
    restaurant.set_step_free_access_score(0);
    restaurant.set_toilet_score(0);
    restaurant.set_parking_score(0);
    restaurant.set_total_score(0);
    restaurant.set_rating("0");
    return restaurant;
}
```

Testing

[ ]:
```
Scanner scanner = new Scanner(System.in);
System.out.println("What is the name of the restaurant? ");
String restaurant_name = scanner.nextLine();
Restaurant restaurant = create_restaurant_record(restaurant_name);
String name = restaurant.get_name();
System.out.println(name);
```

0.11 Method name

set_rating ## What it does Get the users information and stores it in differen record field and sets the rating based on the total rating ## Implementation (how it works) This method takes the scores and the object as the argument. Uses the setters of the object to set the scores to individual attributes. Uses get methods to get the different scores and adds up all the scores. Uses if statement to set the ratings based on hte total score.

[ ]:
```
public static void set_rating (Restaurant restaurant,int␣
↪step_free_access_score, int toilet_score, int parking_score){
 restaurant.set_step_free_access_score(step_free_access_score);
 restaurant.set_toilet_score(toilet_score);
 restaurant.set_parking_score(parking_score);
 int total_score = restaurant.get_step_free_access_score() + restaurant.
↪get_toilet_score() + restaurant.get_parking_score();
 restaurant.set_total_score(total_score);
 if (total_score==9){
   restaurant.set_rating("OUTSTANDING");
 }
 else if (total_score>5){
   restaurant.set_rating("Good");
 }
 else if (total_score<=5){
   restaurant.set_rating("Poor");
 }
 return;
}
```

Testing

[ ]:
```
Restaurant restaurant = create_restaurant_record("taher");
Scanner scanner = new Scanner(System.in);
System.out.println("What is the score for step-free access?");
int step_free_access_score = scanner.nextInt();

System.out.println("What is the score for disabled toilets?");
int toilet_score = scanner.nextInt();

System.out.println("What is the score for disabled parking?");
int parking_score = scanner.nextInt();

set_rating(restaurant, step_free_access_score, toilet_score, parking_score);
System.out.println(restaurant.get_rating());
```

0.12 Method name

print_info ## What it does A further methods should be provided to convert the record to a String in the standard form : "Posh Nosh has a disability rating of GOOD." Printing ratings should be

done independently by calling this method. ## Implementation (how it works) This method takes the object as the argument and uses the getter method to get the name and rating and ouputs them.

[ ]:
```
public static void print_info (Restaurant restaurant){
  System.out.println(restaurant.get_name() + "has a disability rating of " +␣
 ↪restaurant.get_rating()); //
}
```

Testing

[ ]:
```
Restaurant restaurant = create_restaurant_record("taher");
print_info(restaurant);
```

0.13 The complete program

This version will only compile here. To run it copy it into a file called initials.java on your local computer and compile and run it there.

[ ]:
```
/* ************************************
  @author    TAHER AHMED
  @SID       230288917
  @date      26 September 2020
  @version   1
  @desciption
**************************************/
class Restaurant {
  private String name;
  private int step_free_access_score;
  private int toilet_score;
  private int parking_score;
  private int total_score;
  private String rating;
  // Accessor method to get the restaurant's name
  public String get_name() {
      return name;
  }

  // Accessor method to get the step_free_access_score
  public int get_step_free_access_score() {
      return step_free_access_score;
  }

  // Accessor method to get the toilet_score
  public int get_toilet_score() {
      return toilet_score;
  }
```

```
  // Accessor method to get the parking_score
  public int get_parking_score() {
      return parking_score;
  }

  // Accessor method to get the total_score
  public int get_total_score() {
      return total_score;
  }

  // Accessor method to get the rating
  public String get_rating() {
      return rating;
  }
  public void set_name(String name) {
      this.name = name;
  }

  public void set_step_free_access_score(int step_free_access_score) {
      this.step_free_access_score = step_free_access_score;
  }

  public void set_toilet_score(int toilet_score) {
      this.toilet_score = toilet_score;
  }

  public void set_parking_score(int parking_score) {
      this.parking_score = parking_score;
  }

  public void set_total_score(int total_score) {
      this.total_score = total_score;
  }

  public void set_rating(String rating) {
      this.rating = rating;
  }
}

public Restaurant create_restaurant_record(String name) {
  Restaurant restaurant = new Restaurant();
  restaurant.set_name(name);
  // set all the scores to zero
  restaurant.set_step_free_access_score(0);
  restaurant.set_toilet_score(0);
  restaurant.set_parking_score(0);
  restaurant.set_total_score(0);
```

```
  restaurant.set_rating("0");
  return restaurant;
}
public static void set_rating (Restaurant restaurant,int␣
 ↪step_free_access_score, int toilet_score, int parking_score){
    restaurant.set_step_free_access_score(step_free_access_score);
    restaurant.set_toilet_score(toilet_score);
    restaurant.set_parking_score(parking_score);
    int total_score = restaurant.get_step_free_access_score() + restaurant.
 ↪get_toilet_score() + restaurant.get_parking_score();
    restaurant.set_total_score(total_score);
    if (total_score==9){
      restaurant.set_rating("OUTSTANDING");
    }
    else if (total_score>5){
      restaurant.set_rating("GOOD");
    }
    else if (total_score<=5){
      restaurant.set_rating("POOR");
    }
    return;
}
public static void print_info (Restaurant restaurant){
  System.out.println(restaurant.get_name() + " has a disability rating of " +␣
 ↪restaurant.get_rating()); //
}

 // Input only alphabetic characters.
  // This outputs a message then takes an input, if the input does not consist␣
 ↪of alphabetic characters then the user is asked again to input only␣
 ↪alphabetic characters
  public static String get_string_input (String message){
    Scanner input = new Scanner(System.in);
    System.out.println(message);
    String string_input = input.nextLine();

    while (!string_input.matches("^[a-zA-Z]*$")) { // Matches method which␣
 ↪returns true if the each value matches with one of the value in the array of␣
 ↪alpahbetic characters
      System.out.println(message);

      // Check if the input consists of alphabetic characters
      if (!string_input.matches("^[a-zA-Z]*$")) {
        System.out.println("Enter alphabetic characters only!");
      }

      string_input = input.nextLine();
```

```
    }
    return string_input;
  }

   // Input only postive whole number
 public static int get_integer_input(String message) {
  Scanner input = new Scanner(System.in);
  System.out.println(message);
  String str_input = input.nextLine();

  while(!str_input.matches("^-?\\d+$")){ // if the input is not an integer
    System.out.println("Invalid input. Please enter a whole number:");
    str_input = input.nextLine(); // Take input again
  }

  int integer_input = Integer.parseInt(str_input);

  while (integer_input < 0) { // If the value is negative
    System.out.println("Enter a non-negative whole number!");
    str_input = input.nextLine();
    integer_input = Integer.parseInt(str_input);
  }

  return integer_input;
}

public static void main (String[] args){
    Scanner scanner = new Scanner(System.in);
    do {
        String restaurant_name = get_string_input("What is the name of the␣
 ↪restaurant?");
        Restaurant restaurant = create_restaurant_record(restaurant_name);
        int step_free_access_score = get_integer_input("What is the score for␣
 ↪step-free access?");
        int toilet_score = get_integer_input("What is the score for disabled␣
 ↪toilets?");
        int parking_score = get_integer_input("What is the score for disabled␣
 ↪parking?");
        set_rating(restaurant, step_free_access_score, toilet_score,␣
 ↪parking_score);
        print_info(restaurant);
        System.out.println("Another (y/n)?");
    } while (!scanner.nextLine().equals("n"));

    scanner.close();
}
```

[2]:
```
main(null);
```

```
What is the name of the restaurant?
What is the score for step-free access?
What is the score for disabled toilets?
What is the score for disabled parking?
Posh Nosh has a disability rating of GOOD
Another (y/n)?
What is the name of the restaurant?
What is the score for step-free access?
What is the score for disabled toilets?
What is the score for disabled parking?
taher has a disability rating of GOOD
Another (y/n)?
```

END OF LITERATE DOCUMENT