# Auto Mart

### Scenario:

Auto Mart is a renowned Audi dealership located in the city. With a mission to become the go-to destination for car shoppers in the region, they are dedicated to deliver a superior customer experience, from initial inquiry to final purchase. To further enhance this experience, they offer seasonal discounts to customers. To effectively manage and calculate these discounted prices, they require a dedicated software solution.

As their software consultant, you can help them by developing a C# application

### Functionalities:

In class **Car**, implement the below-given properties.

| Datatype | Property Name |
|----------|---------------|
| string | Model |
| string | BodyStyle |
| double | Price |

In class **CarUtility**, implement the below-given methods and also **Inherit** class **Car**.

| Method | Description |
|--------|-------------|
| public bool ValidateCarModel() | This method is used to validate the cake model.<br><br>If the cake model is **A3** or **A8** or **Q5**, then |

In class **CarUtility**, implement the below-given methods and also **Inherit** class **Car**.

| Method | Description |
|---|---|
| public bool ValidateCarModel() | This method is used to validate the cake model.<br><br>If the cake model is **A3** or **A8** or **Q5**, then return **true**. Otherwise return **false.**<br><br>Also display **Invalid car model** in Main method. |
| public Car PriceCalculation() | This method is used to calculate cost of the car after the discount. Set the price after calculation to the property **Price**.<br><br>Then store the **Id, Model, BodyStyle** and **Price after the discount** in **Car** object and return it.<br><br>The price calculation procedure is given below. |

**Formula :**

Price After discount = Price - Discount

| BodyStyle | Discount Percentage |
|---|---|
| Sedan | 5% of Price |

**Formula :**

Price After discount = Price - Discount

| BodyStyle | Discount Percentage |
|-----------|---------------------|
| Sedan | 5% of Price |
| SUV | 10% of Price |
| Others | 0% of Price |

**Note:**

**Car Model** and **BodyStyle** is **Case-sensitive**.

In **Program** class - **Main** method,

1. Get the **Model** from the user.

2. Call the **ValidateCarModel** method, If it returns true then get the **BodyStyle**, and **Price** value from the **user** and move on to step 3, If it returns false then display **invalid car model**

3. Use the values in method **PriceCalculation**

**Note:**

- Keep the properties, methods and classes as **public**.
- Please read the method rules **clearly**.
- Do not use **Environment.Exit()** to terminate the program.
- Do not change the given code template.

Qualifier Assessment

KBA

C #

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

Practice editor

Pritam Biswas

**Note:**

- Keep the properties, methods and classes as **public.**
- Please read the method rules **clearly.**
- Do not use **Environment.Exit()** to terminate the program.
- Do not change the given code template.

**Sample Input 1:**

Enter the model

A3

Enter the body style

Sedan

Enter the price

1500000

**Sample Output 1:**

Model : A3

Body Style : Sedan

Price After Discount : 1425000

**Sample Input 2:**

Enter the model

- Qualifier Assessment
- KBA
- C #
- ANSI SQL
- Dashboard
- Calendar
- Help Desk
- FAQs
- Discussion Community
- Eligibility Status
- Practice editor

A3

Enter the body style

**Sedan**

Enter the price

**1500000**

**Sample Output 1:**

Model : A3

Body Style : Sedan

Price After Discount : 1425000

**Sample Input 2:**

Enter the model

**B7**

**Sample Output 2:**

Invalid car model

30°C Haze

Type here to search

Qualifier Assessment

KBA

C#

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

Practice editor

🏠 Dashboard / Qualifier Assessment / C # / Vitality Health Care

**Description**    Code Editor

## Vitality Health Care

**Scenario:**

A large multispecialty hospital, Vitality Health Care, has recently started administering COVID-19 vaccines to its patients. The hospital management needs an application that can help them analyse and manipulate the booking details for the vaccine doses. The application should be able to track the number of doses available for each vaccine type and the number of doses that have been administered.

As their software consultant, you can help them by developing a C# application.

**Functionalities:**

In class **Vaccine,** implement the below-given properties.

| Data Type | Properties |
|-----------|-----------|
| string | BookingId |
| string | Name |
| string | VaccineType |

🔍 Type here to search    🔵 cognizant    30°C  Haze    2:13 P  4/5/20

Qualifier Assessment

KBA

C#

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

</> Practice editor

**Functionalities:**

In class **Vaccine**, implement the below-given properties.

| Data Type | Properties |
|-----------|------------|
| string | BookingId |
| string | Name |
| string | VaccineType |
| string | DoseNumber |
| string | BookingDate |

In class **Program**, implement the below-given method.

**public static List<Vaccine> VaccineList** -In the code template, it is already provided.

implement the features listed below.

| Method | Description |
|--------|-------------|
| public void AddVaccineDetails (string [] vaccineDetails) | This method is used to add the vaccine booking details to the **VaccineList**. <br><br> This method should separate the values in each string by a **comma** from the **string array** passed as an argument and |

| Method | Description |
|--------|-------------|
| public void AddVaccineDetails (string [] vaccineDetails) | This method is used to add the vaccine booking details to the **VaccineList**.<br><br>This method should separate the values in each string by a **comma** from the **string array** passed as an argument and convert it into a Vaccine object and each **Vaccine** object should be stored in a **VaccineList**. |
| public List<Vaccine> ViewBookingDetailsByDoseNumber(string doseNumber) | This method is used to find the booking details based on the **doseNumber** passed as an argument.<br><br>If the dose number is available in the **VaccineList** then store the booking details as **List** of Vaccine and return it.<br><br>If the dose number is not available in the **VaccineList**, then return an empty **List** and print 'Does number not found' in Main method. |
| public List<Vaccine> ViewBookingDetailsByVaccineType(string vaccineType) | This method is used to find the booking details based on the **vaccineType** passed as an argument.<br><br>If the vaccine type is available in the **VaccineList**, then store the booking details as **List** of Vaccine and return it.<br><br>If the vaccine type is not available in the **VaccineList**, then return an empty **List** and print "Vaccine type not found" in Main method. |

In **Program** class, **Main** method,

1. Get the values from the **user**.

2. Call the methods accordingly and display the result.

3. In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the remaining text represents the output.

**Note:**

- Keep the method and class as **public.**
- Please read the method rules **clearly**.
- Do not use **Environment.Exit()** to terminate the program.
- Do not change the given code template.

**Sample Input / Output:**

1. Add Vaccine Details

2. View Details By Dose Number

3. View Details By Vaccine Type

4. Exit

Enter the choice

1

Enter the number of entries

3

Qualifier Assessment

KBA

C#

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

Practice editor

M001, Raj, COVIDSHIELD, DOSE 1, 25/12/2022

M002, Alex, COVAXIN, DOSE 1, 25/12/2022

M003, Mohammed, COVIDSHIELD, DOSE 2, 25/12/2022

1. Add Vaccine Details

2. View Details By Dose Number

3. View Details By Vaccine Type

4. Exit

Enter the choice

2

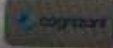Enter the dose number

DOSE 1

M001 Raj COVIDSHIELD DOSE 1 25/12/2022

M002 Alex COVAXIN DOSE 1 25/12/2022

1. Add Vaccine Details

2. View Details By Dose Number

3. View Details By Vaccine Type

4. Exit

Qualifier Assessment

KBA

C #

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

</> Practice editor

Pritam Biswas

4. Exit

Enter the choice

2

Enter the dose number

**DOSE 4**

Does number not found


1. Add Vaccine Details

2. View Details By Dose Number

3. View Details By Vaccine Type

4. Exit

Enter the choice

3

Enter the vaccine type

**COVIDSHIELD**

M001 Raj COVIDSHIELD DOSE 1 25/12/2022

M003 Mohammed COVIDSHIELD DOSE 2 25/12/2022


1. Add Vaccine Details

2. View Details By Dose Number

4. Exit

Enter the choice

3

Enter the vaccine type

**COVIDSHIELD**

M001 Raj COVIDSHIELD DOSE 1 25/12/2022

M003 Mohammed COVIDSHIELD DOSE 2 25/12/2022

1. Add Vaccine Details

2. View Details By Dose Number

3. View Details By Vaccine Type

4. Exit

Enter the choice

4

Thank you.

# Phoenix Club

**Scenario:**

The Phoenix cricket coaching club's management has an online application that allows aspiring cricket players to apply for admission. To improve their services and streamline their operations, the management wants to implement a feature that allows them to analyze and manipulate player details based on age and interests.

You being their software consultant, help them by developing a C# application.

**Functionalities:**

In class **Player**, implement the below-given properties.

| Data Type | Properties |
|-----------|------------|
| string | Number |
| string | Name |
| int | Age |
| string | InterestedIn |

In class **Program**, implement the below-given method.

public static List<Player> PlayerList -In the code template, it is already provided.

In class **Program**, implement the below-given method.

**public static List<Player> PlayerList** -In the code template, it is already provided.

implement the features listed below.

| Method | Description |
|---|---|
| public void AddPlayerDetails (string[ ] playerDetails) | This method is used to add the player details into the **PlayerList.**<br><br>This method should separate the values in each string by a **colon** (:) from the input array and assign those values to the **Player** object, then add each object to the **PlayerList.** |
| public int FindCountOfPlayersByAge (int age) | This method is used to find the **count** of the players whose age is less than the age at which an argument is passed and return the **count**.<br><br>If the player does **not** have an **age** less than the age passed as an argument, then return **0**. |
| public List<Player> ViewDetailsByInterestedIn(string interestedIn) | This method is used to find the player details based on the **interestedIn** passed as an argument.<br><br>If the **interestedIn** is found in the **PlayerList** , then add those player details to the List and return that **List**, else, return an **empty list.** |

In **Program** class, **Main** method,

In **Program** class, **Main** method.

1. Get the values from the **user**.

2. Call the methods accordingly and display the result.

 — If the **FindCountOfPlayersByAge** method returns 0, then display **"Player not found"**.

 — If the **ViewDetailsByInterestedIn** method returns an empty list, then display **"Player not found"**.

3. In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the remaining text represents the output.

**Note:**

- Keep the method and class as **public.**
- Please read the method rules **clearly**.
- Do not use **Environment.Exit()** to terminate the program.
- Do not change the given code template.

**Sample Input and Output :**

1. Add Player Details

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

**Sample Input and Output :**

1. Add Player Details

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

1

Enter the number of entries

4

**AP36:Abraham:20:All Rounder**

**AP37:Adam:15:Bowler**

**AP38:Mathew:14:Batter**

**AP40:Leo:19:Bowler**

1. Add Player Details

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

Qualifier Assessment

KBA

C #

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

</> Practice editor

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

2

Enter the age

14

Player not found

1. Add Player Details

2. Find Count By Age

3. View Details By interested In

4. Exit

Enter the choice

2

Enter the age

19

The players below 19 is 2

Qualifier Assessment

KBA

C#

ANSI SQL

Dashboard

Calendar

Help Desk

FAQs

Discussion Community

Eligibility Status

Practice editor

1. Add Player Details

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

3

Enter the interested in

**Wicket Keeper**

Player not found

1. Add Player Details

2. Find Count By Age

3. View Details By Interested In

4. Exit

Enter the choice

3

Enter the interested in

**Bowler**

Enter the choice

**3**

Enter the interested in

**Bowler**


AP37  Adam  15  Bowler

AP40  Leo    19  Bowler


1. Add Player Details

2. Find Count By Age

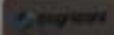3. View Details By Interested In

4. Exit

Enter the choice

**4**

Thank you.

Powered by Tekstac