Mehmet Hüsna Kışla

Soru 7:

Yöntem 1:

public class Car

{

public Car(string make)

{

Make = make;

}

public string Make { get; set; }

}

Yöntem 2:

public class Car

{

public Car(string Make)

{

this.Make = Make;

}

public string Make { get; set; }

}

Yöntem 3:

public class Car

{

public Car(string make)

{

\_make = make;

}

private string \_make;

public string Make { get { return \_make; } set { \_make = Make; } }

}

Soru 11:

foreach (SoftwareSpecialist item in team)

{

if (item is SoftwareTester)

{

SoftwareTester tester = item as SoftwareTester;

tester.TesterOzellik = "Tester Özellik";

}

}

Soru 12:

public enum GenderType

{

Erkek, Kadin

}

public class Person

{

public string FirstName;

public string LastName;

public DateTime DateOfBirth;

public GenderType Gender;

}

Soru 13:

public class Person

{

public string FirstName;

public string LastName;

public DateTime DateOfBirth;

public string Gender;

public Person()

{

int result = DateTime.Compare(DateOfBirth, DateTime.Now);

if(result == -1)

Console.WriteLine("Bu değer girilemez");

}

}

Soru 14:

public class Person

{

public string FirstName;

public string LastName;

public DateTime DateOfBirth;

public GenderType Gender;

public Person(string firstName, string lastName)

{

FirstName = firstName;

LastName = lastName;

}

}

Soru 15:

public class Seat

{

public Seat(int number)

{

Number = number;

}

public int Number { get; }

public Person Passenger { get; set; }

}

Soru 16:

private Person \_passenger;

public Person Passenger

{

get

{

return \_passenger;

}

set

{

if (Passenger != null)

Console.WriteLine("Koltuk zaten dolu !");

else

\_passenger = value;

}

}

Soru 17:

public enum KoltukDurum

{

Bos, Dolu

}

public class Seat

{

public Seat(int number)

{

Number = number;

}

public int Number { get; }

private Person \_passenger;

public Person Passenger

{

get

{

return \_passenger;

}

set

{

if (Passenger != null)

Console.WriteLine("Koltuk zaten dolu !");

else

\_passenger = value;

}

}

public KoltukDurum Durum { get; set; }

}

Soru 18:

public class Bus

{

public Bus(int seatCount)

{

Seats = new List<Seat>();

for (int i = 0; i < seatCount; i++)

{

int seatNumber = i + 1;

Seat seat = new Seat(seatNumber);

Seats.Add(seat);

}

}

public Person Driver { get; set; }

public List<Seat> Seats { get; set; }

}

Soru 19:

public class Bus

{

public readonly int koltukSayisi;

public Bus(int seatCount)

{

Seats = new List<Seat>();

koltukSayisi = seatCount;

for (int i = 0; i < koltukSayisi; i++)

{

int seatNumber = i + 1;

Seat seat = new Seat(seatNumber);

Seats.Add(seat);

}

}

public Person Driver { get; set; }

public List<Seat> Seats { get; set; }

}

Soru 20:

public class Seat

{

public Seat(int number)

{

Number = number;

}

public int Number { get; }

}

Soru 21:

public enum KoltukTipi

{

Erkek, Kadin

}

public class Seat

{

public Seat(int number)

{

Number = number;

}

public int Number { get; }

public KoltukTipi KoltukTipi { get; set; }

}

**PROJE DOSYALARI**

**Bus.cs**

public class Bus

{

public readonly int koltukSayisi;

public Bus(int seatCount)

{

Seats = new List<Seat>();

koltukSayisi = seatCount;

for (int i = 0; i < koltukSayisi; i++)

{

int seatNumber = i + 1;

Seat seat = new Seat(seatNumber);

Seats.Add(seat);

}

}

public Person Driver { get; set; }

public List<Seat> Seats { get; set; }

}

**KoltukDurum.cs**

public enum KoltukDurum

{

Bos, Dolu

}

**KoltukTipi.cs**

public enum KoltukTipi

{

Erkek, Kadin

}

**Person.cs**

public class Person

{

}

**Seat.cs**

public class Seat

{

public Seat(int number)

{

Number = number;

}

public int Number { get; }

private Person \_passenger;

public Person Passenger

{

get

{

return \_passenger;

}

set

{

if (Passenger != null)

Console.WriteLine("Koltuk zaten dolu !");

else

\_passenger = value;

}

}

public KoltukDurum Durum { get; set; }

public KoltukTipi KoltukTipi { get; set; }

}