Connor Perron

BCS426

Prof. Aydin

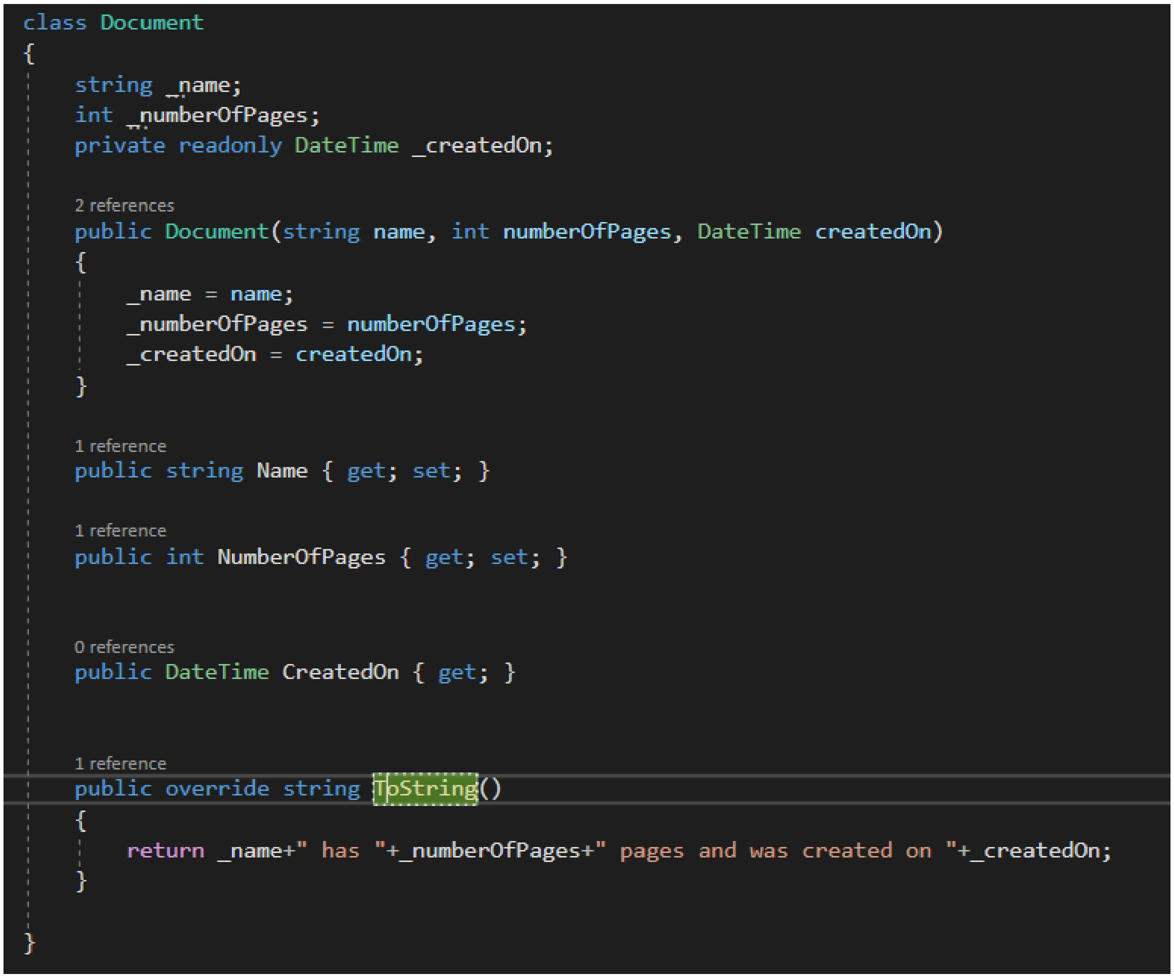
Spring 2021

Question 1: Create a C# class called Document

a) **with the following Properties:**
   i) **Name (string)**
   ii) **NumberOfPages (int)**
   iii) **CreatedOn (DateTime): make this a readonly property. Note that DateTime is a struct in the namespace System**

b) **Add a constructor to allow initialization of all the properties ~~setting properties Name and NumberOfPages front~~**

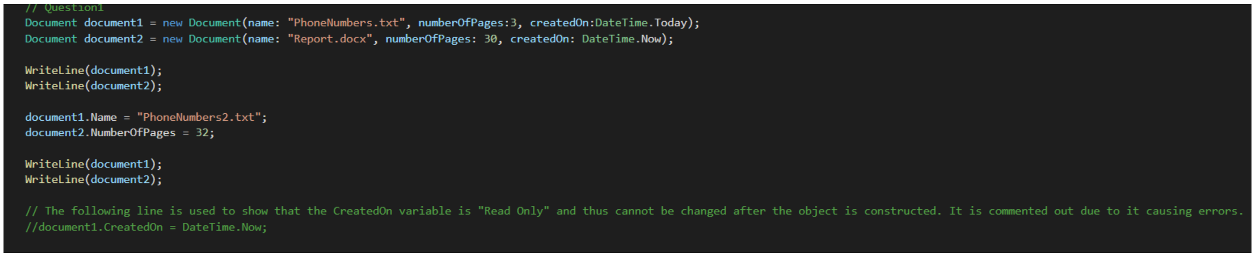
c) **Add ToString Method (hint: override toString method of System.Object class)**



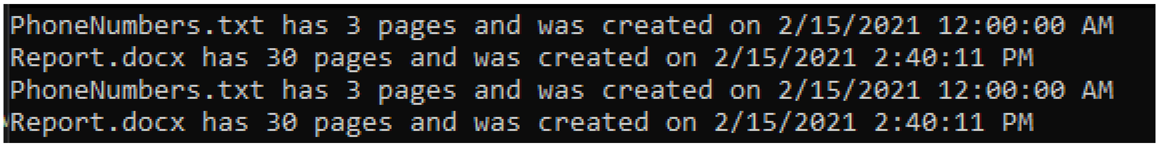
```
class Document
{
    string _name;
    int _numberOfPages;
    private readonly DateTime _createdOn;

    2 references
    public Document(string name, int numberOfPages, DateTime createdOn)
    {
        _name = name;
        _numberOfPages = numberOfPages;
        _createdOn = createdOn;
    }

    1 reference
    public string Name { get; set; }

    1 reference
    public int NumberOfPages { get; set; }


    0 references
    public DateTime CreatedOn { get; }


    1 reference
    public override string ToString()
    {
        return _name+" has "+_numberOfPages+" pages and was created on "+_createdOn;
    }

}
```

Then, write a Main method to test your class Document.

a) **Create document1 with name “PhoneNumbers.txt” and 3 pages**
b) **Create document2 with name “Report.docx” and 30 pages**
c) **Demo how you cannot change the CreatedOn property of each document object**



```
// Question1
Document document1 = new Document(name: "PhoneNumbers.txt", numberOfPages:3, createdOn:DateTime.Today);
Document document2 = new Document(name: "Report.docx", numberOfPages: 30, createdOn: DateTime.Now);

WriteLine(document1);
WriteLine(document2);

document1.Name = "PhoneNumbers2.txt";
document2.NumberOfPages = 32;

WriteLine(document1);
WriteLine(document2);

// The following line is used to show that the CreatedOn variable is "Read Only" and thus cannot be changed after the object is constructed. It is commented out due to it causing errors.
//document1.CreatedOn = DateTime.Now;
```



```
PhoneNumbers.txt has 3 pages and was created on 2/15/2021 12:00:00 AM
Report.docx has 30 pages and was created on 2/15/2021 2:40:11 PM
PhoneNumbers.txt has 3 pages and was created on 2/15/2021 12:00:00 AM
Report.docx has 30 pages and was created on 2/15/2021 2:40:11 PM
```

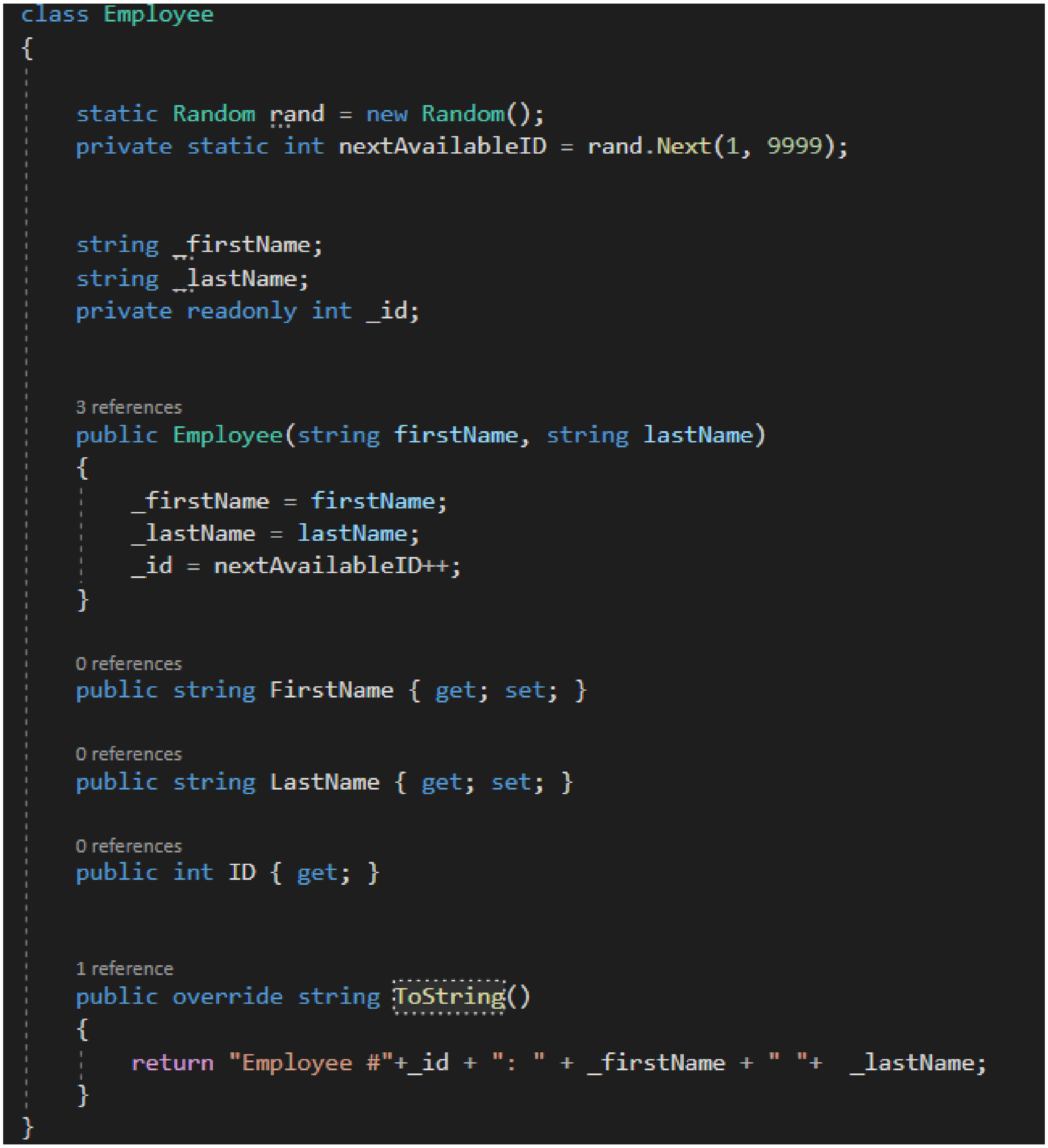
2. Create a class Employee

a) *private* and *static field* nextAvailableID (int): create a *static constructor* to initialize this field to a random number (hint: you can use System.Random class). You can assume that IDs are numbers between 1-9999. Since nextAvailableID is static, all the Employee objects will share this field.

b) with properties

- **FirstName (string)**
- **LastName (string),**
- **ID (int): make it read only to allow initializing only with a constructor and then no more.**

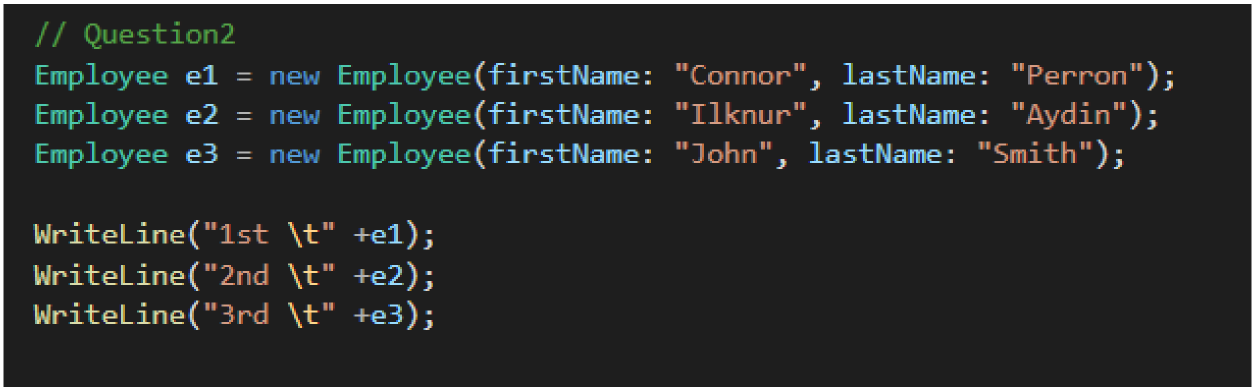
c) Add a constructor to allow setting properties FirstName and LastName. Hint: ID will be initialized to nextAvailableID (make sure to increment nextAvailableID by one)



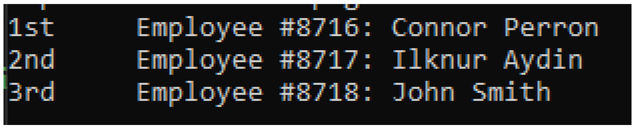
```
class Employee
{

    static Random rand = new Random();
    private static int nextAvailableID = rand.Next(1, 9999);


    string _firstName;
    string _lastName;
    private readonly int _id;


    3 references
    public Employee(string firstName, string lastName)
    {
        _firstName = firstName;
        _lastName = lastName;
        _id = nextAvailableID++;
    }

    0 references
    public string FirstName { get; set; }

    0 references
    public string LastName { get; set; }

    0 references
    public int ID { get; }


    1 reference
    public override string ToString()
    {
        return "Employee #"+_id + ": " + _firstName + " "+  _lastName;
    }
}
```

Then, write a Main method to test your class Employee.

a) **Create four Employee objects with references e1, e2, and e3 by providing first and last name of each employee**
b) **Display the full name and id of each employee - note that each employee will have an automatically generated ID by the class Employee.**



```
// Question2
Employee e1 = new Employee(firstName: "Connor", lastName: "Perron");
Employee e2 = new Employee(firstName: "Ilknur", lastName: "Aydin");
Employee e3 = new Employee(firstName: "John", lastName: "Smith");

WriteLine("1st \t" +e1);
WriteLine("2nd \t" +e2);
WriteLine("3rd \t" +e3);
```



```
1st     Employee #8716: Connor Perron
2nd     Employee #8717: Ilknur Aydin
3rd     Employee #8718: John Smith
```