

Άσκηση 1: Προσθήκη και Αφαίρεση Στοιχείων από Ουρά

Γράψε ένα πρόγραμμα που:

Δημιουργεί μια ουρά και προσθέτει 5 αριθμούς.

Αφαιρεί έναν αριθμό από την ουρά.

Εμφανίζει τα στοιχεία που απομένουν στην ουρά.

Οδηγίες:

Χρησιμοποίησε τη μέθοδο Enqueue() για να προσθέσεις στοιχεία στην ουρά.

Χρησιμοποίησε τη μέθοδο Dequeue() για να αφαιρέσεις το πρώτο στοιχείο.

Λύση:

```
using System;
```

```
using System.Collections.Generic;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        Queue<int> queue = new Queue<int>();
```

```
        // Προσθήκη 5 αριθμών στην ουρά
```

```
        queue.Enqueue(10);
```

```
        queue.Enqueue(20);
```

```
        queue.Enqueue(30);
```

```
queue.Enqueue(40);
queue.Enqueue(50);

// Εκτύπωση της ουράς
Console.WriteLine("Στοιχεία στην ουρά:");
foreach (int number in queue)
{
    Console.WriteLine(number);
}

// Αφαίρεση του πρώτου στοιχείου από την ουρά
int removedElement = queue.Dequeue();
Console.WriteLine("\nΤο στοιχείο που αφαιρέθηκε είναι: " + removedElement);

// Εκτύπωση της ουράς μετά το Dequeue
Console.WriteLine("Στοιχεία στην ουρά μετά το Dequeue:");
foreach (int number in queue)
{
    Console.WriteLine(number);
}
}
```

Άσκηση 2: Εμφάνιση του Πρώτου Στοιχείου Χωρίς Αφαίρεση

Γράψε ένα πρόγραμμα που προσθέτει 5 αριθμούς σε μια ουρά και μετά εμφανίζει το πρώτο στοιχείο χωρίς να το αφαιρέσει από την ουρά.

Οδηγίες:

Χρησιμοποίησε τη μέθοδο `Enqueue()` για να προσθέσεις στοιχεία στην ουρά.

Χρησιμοποίησε τη μέθοδο `Peek()` για να εμφανίσεις το πρώτο στοιχείο.

Λύση:

```
using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        Queue<int> queue = new Queue<int>();

        // Προσθήκη 5 αριθμών στην ουρά
        queue.Enqueue(10);
        queue.Enqueue(20);
        queue.Enqueue(30);
        queue.Enqueue(40);
        queue.Enqueue(50);

        // Εκτύπωση του πρώτου στοιχείου χωρίς αφαίρεση
        int firstElement = queue.Peek();
        Console.WriteLine("Το πρώτο στοιχείο στην ουρά είναι: " + firstElement);
    }
}
```

Άσκηση 3: Διαγραφή Όλων των Στοιχείων από την Ουρά

Γράψε ένα πρόγραμμα που προσθέτει αριθμούς σε μια ουρά και στη συνέχεια διαγράφει όλα τα στοιχεία.

Οδηγίες:

Χρησιμοποίησε τη μέθοδο `Enqueue()` για να προσθέσεις στοιχεία.

Χρησιμοποίησε τη μέθοδο `Clear()` για να διαγράψεις όλα τα στοιχεία από την ουρά.

Λύση:

```
using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        Queue<int> queue = new Queue<int>();

        // Προσθήκη αριθμών στην ουρά
        queue.Enqueue(10);
        queue.Enqueue(20);
        queue.Enqueue(30);

        // Εκτύπωση της ουράς πριν τη διαγραφή
        Console.WriteLine("Στοιχεία στην ουρά πριν τη διαγραφή:");
        foreach (int number in queue)
        {
            Console.WriteLine(number);
        }

        // Διαγραφή όλων των στοιχείων από την ουρά
        queue.Clear();
    }
}
```

```
// Εκτύπωση της ουράς μετά τη διαγραφή
Console.WriteLine("Στοιχεία στην ουρά μετά τη διαγραφή:");
Console.WriteLine("Η ουρά έχει " + queue.Count + " στοιχεία.");
}
}
```

Άσκηση 4: Ουρά Πελατών σε Κατάστημα

Γράψε ένα πρόγραμμα που διαχειρίζεται μια ουρά πελατών σε ένα κατάστημα. Το πρόγραμμα πρέπει:

Να προσθέτει πελάτες στην ουρά.

Να αφαιρεί πελάτες που εξυπηρετήθηκαν.

Να εμφανίζει την ουρά των πελατών μετά από κάθε ενέργεια.

Οδηγίες:

Χρησιμοποίησε τη μέθοδο Enqueue() για να προσθέσεις πελάτες στην ουρά.

Χρησιμοποίησε τη μέθοδο Dequeue() για να αφαιρέσεις πελάτες που εξυπηρετήθηκαν.

Λύση:

```
using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        Queue<string> customers = new Queue<string>();
```

```
while (true)
{
    Console.WriteLine("1. Προσθήκη Πελάτη");
    Console.WriteLine("2. Εξυπηρέτηση Πελάτη");
    Console.WriteLine("3. Προβολή Πελατών");
    Console.WriteLine("4. Έξοδος");
    Console.Write("Επιλογή: ");
    int choice = int.Parse(Console.ReadLine());

    if (choice == 1)
    {
        Console.Write("Εισάγετε το όνομα του πελάτη: ");
        string customerName = Console.ReadLine();
        customers.Enqueue(customerName);

        Console.WriteLine("Ο πελάτης " + customerName + " προστέθηκε στην ουρά.");
    }
    else if (choice == 2)
    {
        if (customers.Count > 0)
        {
            string servedCustomer = customers.Dequeue();
            Console.WriteLine("Ο πελάτης " + servedCustomer + " εξυπηρετήθηκε.");
        }
        else
        {
            Console.WriteLine("Δεν υπάρχουν πελάτες στην ουρά.");
        }
    }
    else if (choice == 3)
    {
        Console.WriteLine("Πελάτες στην ουρά:");
    }
}
```

```
        foreach (string customer in customers)
        {
            Console.WriteLine(customer);
        }
    }
    else if (choice == 4)
    {
        break;
    }

    Console.WriteLine(); // Κενή γραμμή για καλύτερη εμφάνιση
}
}
```

Project: Ουρά Εκτυπωτή

Δημιούργησε ένα πρόγραμμα που διαχειρίζεται μια ουρά εκτυπωτή. Κάθε φορά που κάποιος στέλνει ένα έγγραφο για εκτύπωση, αυτό προστίθεται στην ουρά. Ο εκτυπωτής εξυπηρετεί το πρώτο έγγραφο που μπήκε στην ουρά.

Το πρόγραμμα πρέπει να επιτρέπει:

Προσθήκη εγγράφου στην ουρά εκτύπωσης.

Εξυπηρέτηση του πρώτου εγγράφου.

Προβολή των εγγράφων που παραμένουν στην ουρά εκτύπωσης.

Οδηγίες:

Χρησιμοποίησε τη μέθοδο `Enqueue()` για να προσθέσεις έγγραφα.

Χρησιμοποίησε τη μέθοδο `Dequeue()` για να αφαιρέσεις το πρώτο έγγραφο που εκτυπώθηκε.

Λύση:

```
using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        Queue<string> printQueue = new Queue<string>();

        while (true)
        {
            Console.WriteLine("1. Προσθήκη Εγγράφου για Εκτύπωση");
            Console.WriteLine("2. Εκτύπωση Εγγράφου");
            Console.WriteLine("3. Προβολή Ουράς Εκτύπωσης");
            Console.WriteLine("4. Έξοδος");
            Console.Write("Επιλογή: ");
            int choice = int.Parse(Console.ReadLine());

            if (choice == 1)
            {
                Console.Write("Εισάγετε το όνομα του εγγράφου: ");
                string documentName = Console.ReadLine();
                printQueue.Enqueue(documentName);

                Console.WriteLine("Το έγγραφο " + documentName + " προστέθηκε στην ουρά εκτύπωσης.");
            }
            else if (choice == 2)
            {
                if (printQueue.Count > 0)
```

```

    {
        string printedDocument = printQueue.Dequeue();
        Console.WriteLine("Το έγγραφο " + printedDocument + " εκτυπώθηκε.");
    }
    else
    {
        Console.WriteLine("Η ουρά εκτύπωσης είναι κενή.");
    }
}
else if (choice == 3)
{
    if (printQueue.Count > 0)
    {
        Console.WriteLine("Έγγραφα στην ουρά εκτύπωσης:");
        foreach (string document in printQueue)
        {
            Console.WriteLine(document);
        }
    }
    else
    {
        Console.WriteLine("Δεν υπάρχουν έγγραφα στην ουρά εκτύπωσης.");
    }
}
else if (choice == 4)
{
    break;
}

Console.WriteLine(); // Κενή γραμμή για καλύτερη εμφάνιση
}
}

```

}