Queue On Steroids

LIGHTNING TALK 2017-09-28

BY ARNOLD BARNA

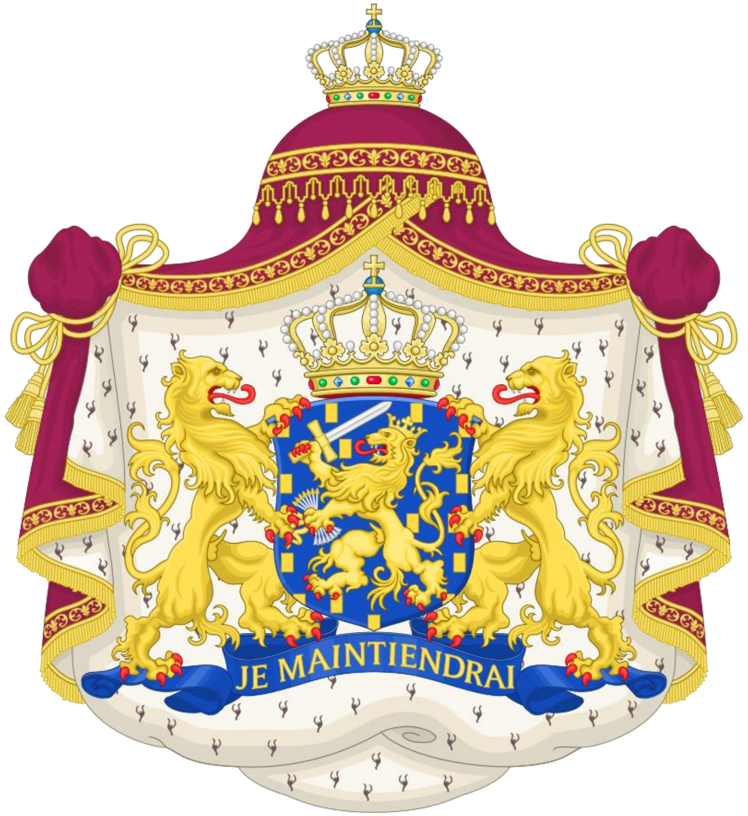
Basic Facts

- System.Collections is boring
- Our knowledge on object-oriented approach is growing
- We are hungry for more …
- And Chuck Norris sees us all

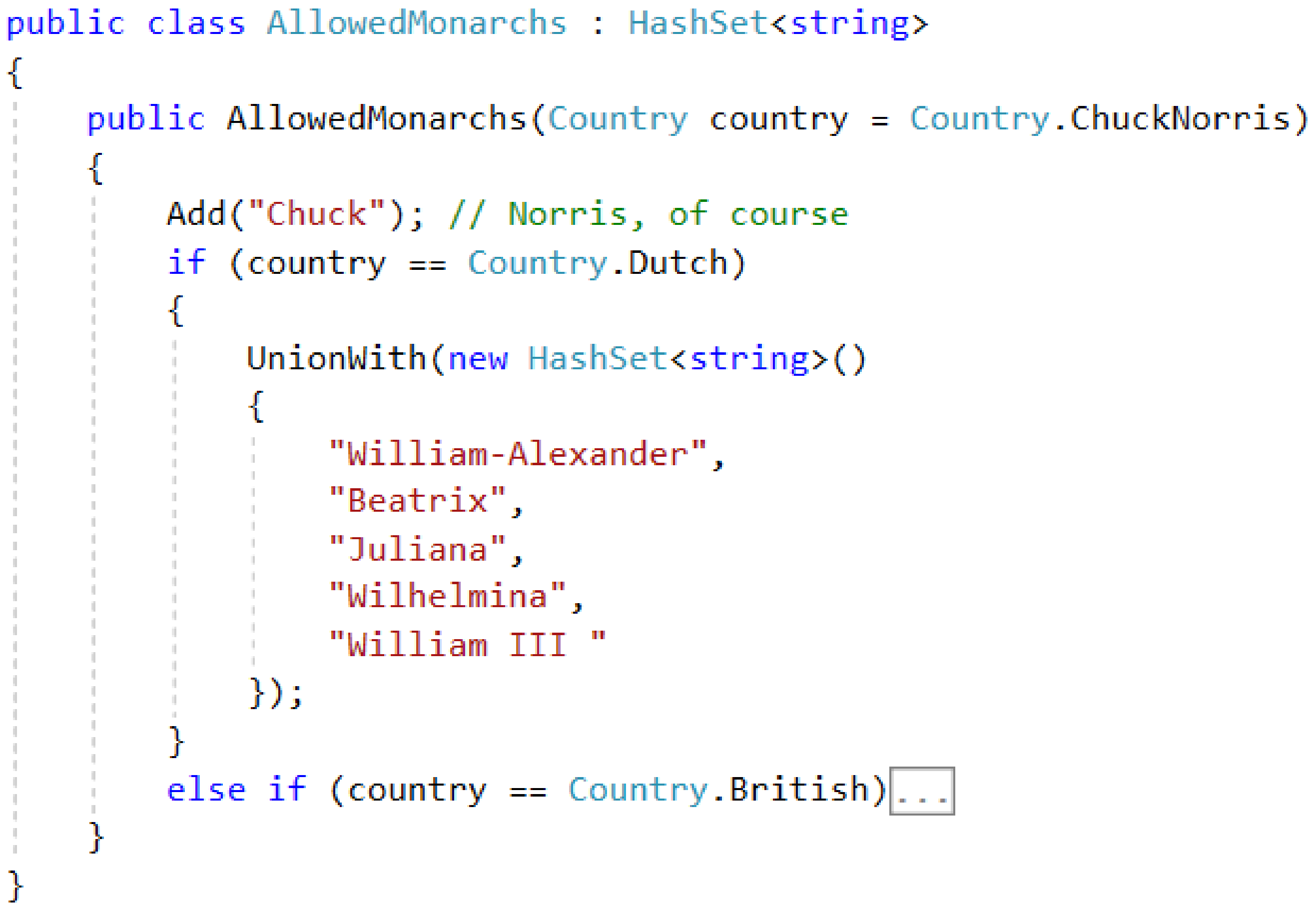
Let’s create our special smart Queue

- ***Queue**: first-in, first-out (Enqueue / Dequeue)*

- *Only a chosen few of the **Dutch monarchs** are allowed now in our queue*
  *… and **Chuck**, of course*

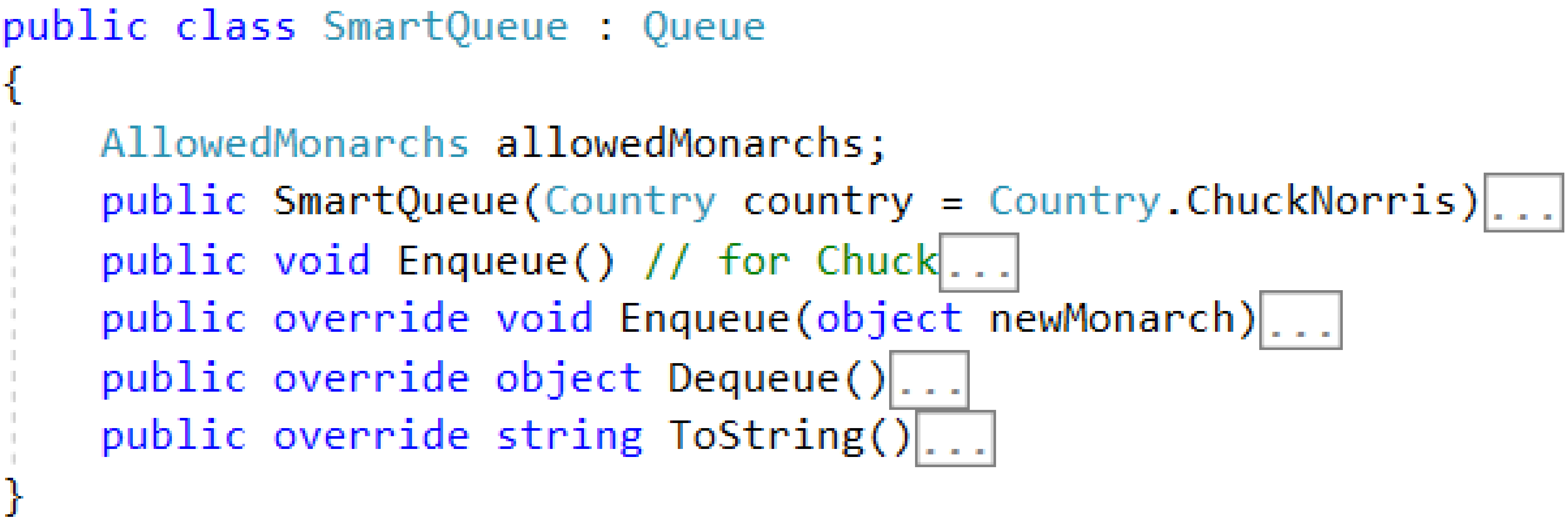


Allowed monarchs



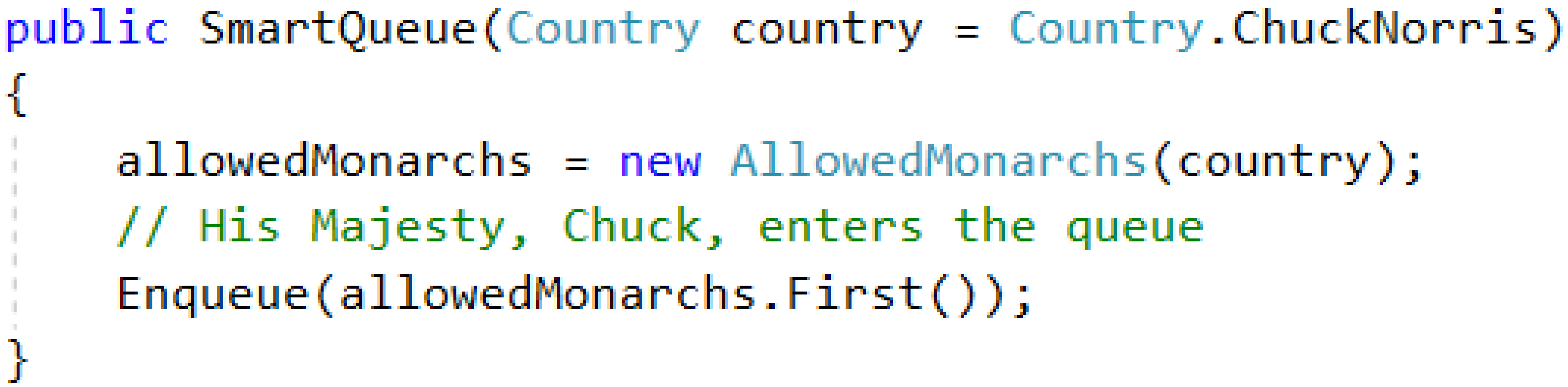
```
public class AllowedMonarchs : HashSet<string>
{
    public AllowedMonarchs(Country country = Country.ChuckNorris)
    {
        Add("Chuck"); // Norris, of course
        if (country == Country.Dutch)
        {
            UnionWith(new HashSet<string>()
            {
                "William-Alexander",
                "Beatrix",
                "Juliana",
                "Wilhelmina",
                "William III "
            });
        }
        else if (country == Country.British)...
    }
}
```

SmartQueue overview



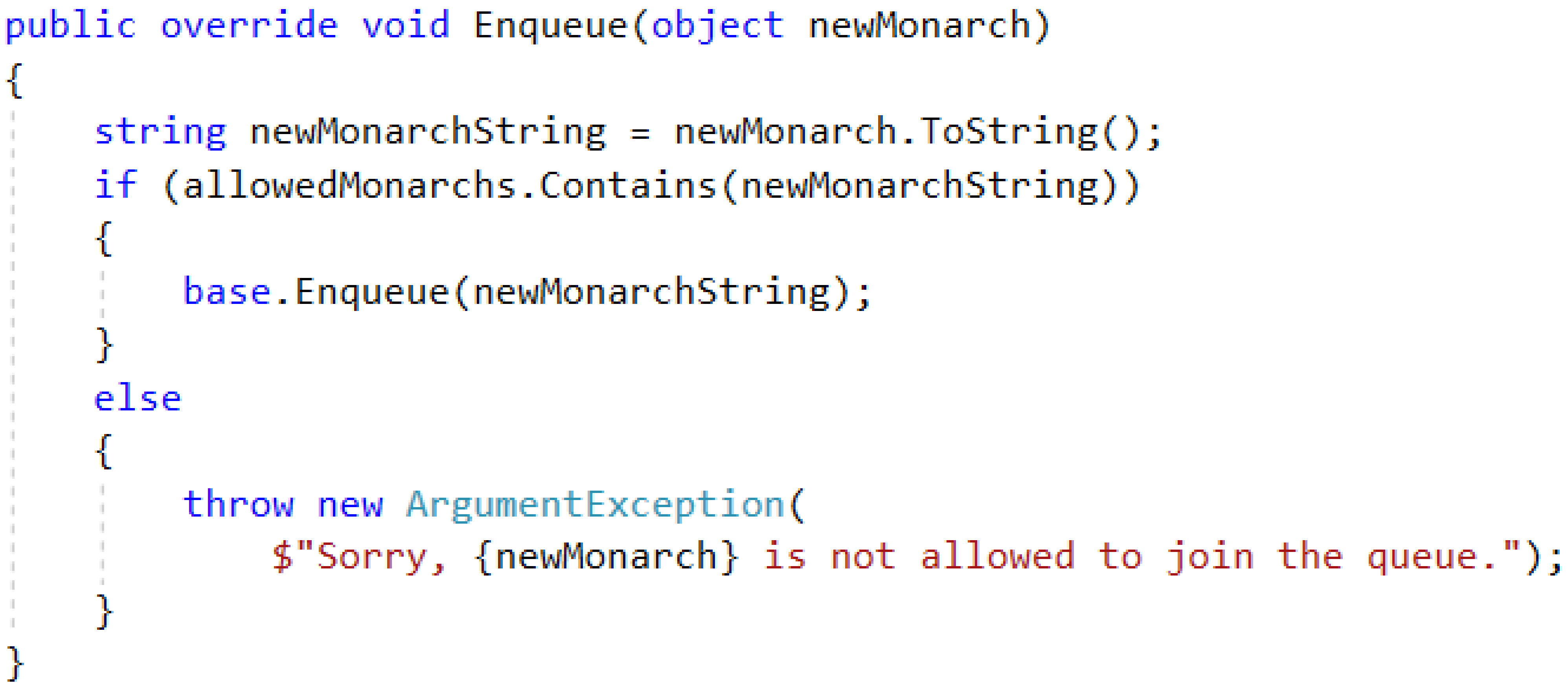
```
public class SmartQueue : Queue
{
    AllowedMonarchs allowedMonarchs;
    public SmartQueue(Country country = Country.ChuckNorris)...
    public void Enqueue() // for Chuck...
    public override void Enqueue(object newMonarch)...
    public override object Dequeue()...
    public override string ToString()...
}
```

Constructor w/ auto parameter



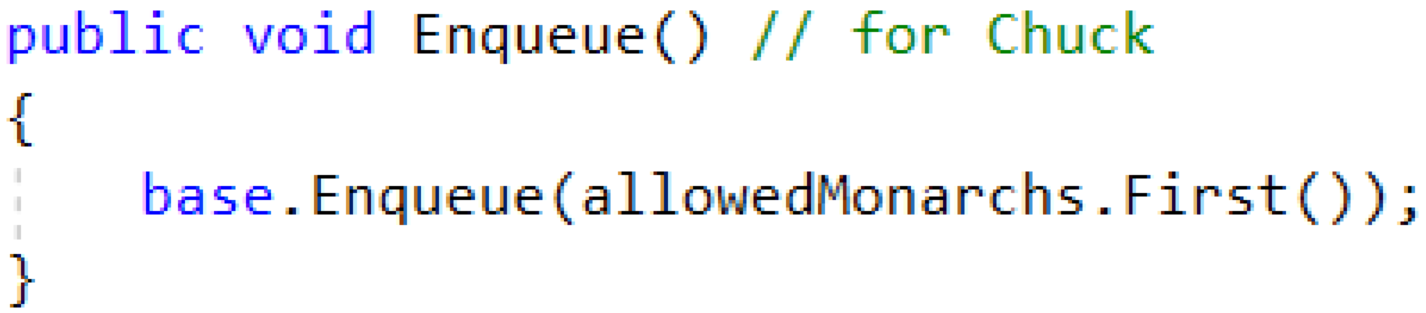
```
public SmartQueue(Country country = Country.ChuckNorris)
{
    allowedMonarchs = new AllowedMonarchs(country);
    // His Majesty, Chuck, enters the queue
    Enqueue(allowedMonarchs.First());
}
```

Enqueue()



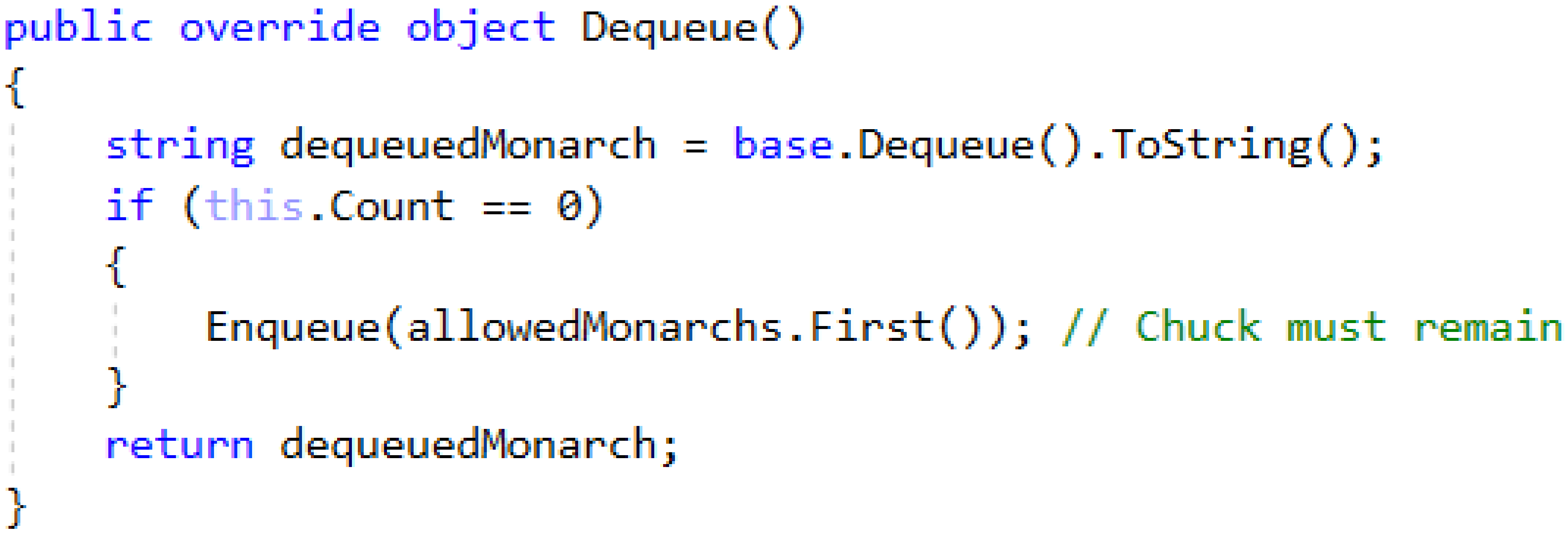
```
public override void Enqueue(object newMonarch)
{
    string newMonarchString = newMonarch.ToString();
    if (allowedMonarchs.Contains(newMonarchString))
    {
        base.Enqueue(newMonarchString);
    }
    else
    {
        throw new ArgumentException(
            $"Sorry, {newMonarch} is not allowed to join the queue.");
    }
}
```

Enqueue()



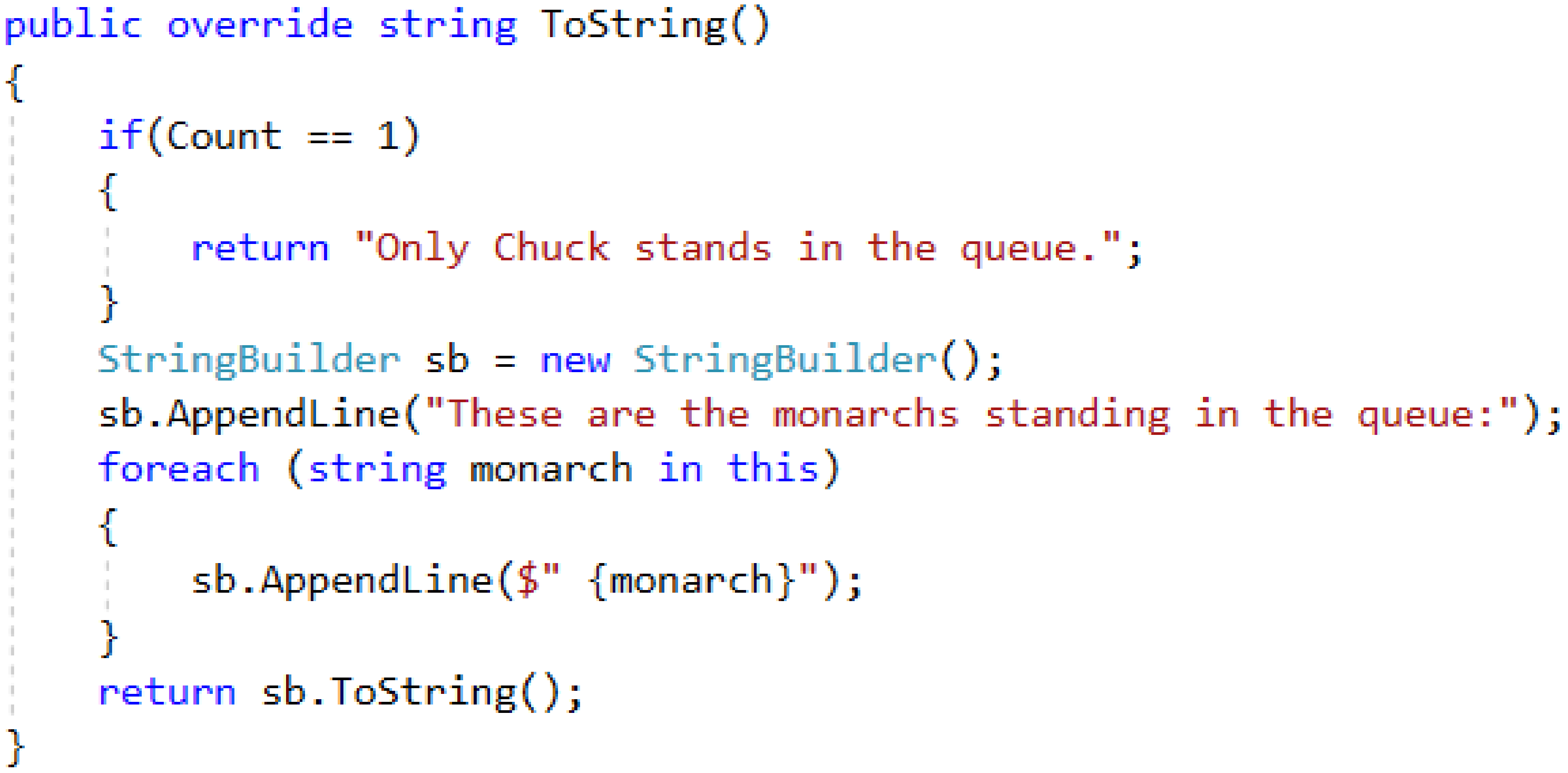
```
public void Enqueue() // for Chuck
{
    base.Enqueue(allowedMonarchs.First());
}
```

Dequeue()



```
public override object Dequeue()
{
    string dequeuedMonarch = base.Dequeue().ToString();
    if (this.Count == 0)
    {
        Enqueue(allowedMonarchs.First()); // Chuck must remain
    }
    return dequeuedMonarch;
}
```

ToString()



```
public override string ToString()
{
    if(Count == 1)
    {
        return "Only Chuck stands in the queue.";
    }
    StringBuilder sb = new StringBuilder();
    sb.AppendLine("These are the monarchs standing in the queue:");
    foreach (string monarch in this)
    {
        sb.AppendLine($" {monarch}");
    }
    return sb.ToString();
}
```

Thank you for the attention

Available on GitHub:

Code:

- https://github.com/greenfox-academy/**bramble100**/tree/master/week-04/**lightning-talk**

Slides:

- https://github.com/greenfox-academy/**bramble100**/tree/master/week-04/**lightning-talk**