



---

# PROXY

---

Design Pattern





**INTERNATIONAL ISLAMIC UNIVERSITY ISLAMABAD**

---

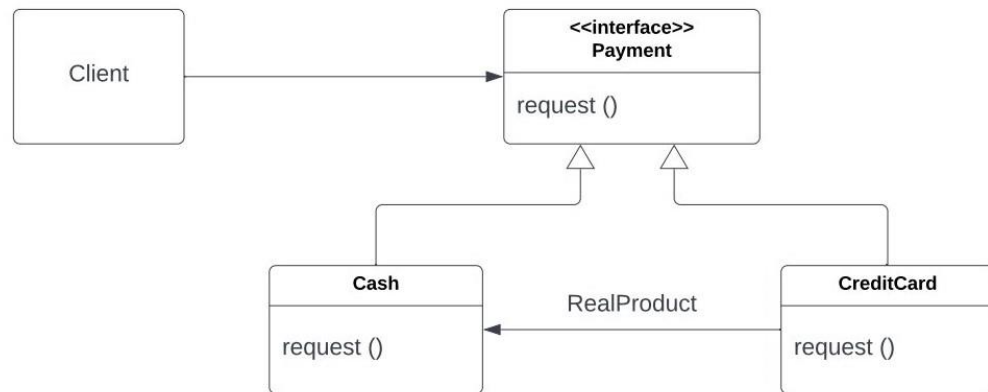
## ***Report***

---

**Name:** Tamoor Ahmad  
**Registration No:** 4125-FBAS/BSSE/F20  
**Course Code:** SE322  
**Course Title:** Software Design and Architecture

**Submitted to:** Dr. Muhammad Nasir

## Class Diagram:



Consider the example of a cheque or credit card that can work as proxy for your bank account. It can be used in place of cash and provides a means of accessing that cash when required.

## Source Code:

```
using System;

namespace bankproxy.structral
{
    /// <summary>
    /// Proxy Design Pattern
    /// </summary>
    public class Program
    {
        public static void Main(string[] args)
        {
            // Create proxy and request a service
        }
    }
}
```

```

        Proxy proxy = new Proxy();
        proxy.Request();
        // Wait for user
        Console.ReadKey();
    }

}

/// <summary>
/// The 'cardit' abstract class
/// </summary>
public abstract class cardit
{
    public abstract void Request();
}

public class providehash : cardit
{
    public override void Request()
    {
        Console.WriteLine("Called providehash.Request()");
    }
}

/// <summary>
/// The 'Proxy' class
/// </summary>
public class Proxy : cardit
{
    private providehash providehash;
    public override void Request()
    {

```

```
// Use 'lazy initialization'
if (providechash == null)
{
    providechash = new providechash();
}
providechash.Request();
}
}
}
```

## Output:

Select C:\Users\ITS\source\repos\Proxy\Proxy\bin\Debug\Proxy.exe

Called providehash.Request()



Type here to search

