Polymorphism in C# - Complete Illustrated Guide

This guide demonstrates interface-based and virtual/override polymorphism using a concise animal hierarchy.

Each code block is shown as a colorhighlighted snippet for readability.

IHayvan.cs

IHayvan interface -- common contract every animal must follow.

```
using System;

public interface IHayvan
{
    // omurgal:lar, memeliler v.s
    public string sinif();
    // hayvanin adi
    public string tur();
    // Yumurtlama dogurma v.s
    public bool uremeYaparMi();
    public bool soyuTukendiMi();
    public string zehirliMi();
}
```

Yunus.cs

Yunus base class implements lHayvan; virtual members allow overriding.

```
using System;
public inaesfăqeu HayVaayvan
    //Vommungalmeaod mekalitemla.gelen ve türetilmiş sınıfta override edilebilen metod.
public viringlsboot() pyuTukendiMi()
     {/ hayvanin adi
    publi¢ %tmingntme∢)i tukenmediginden false dondurecek
     // Ymetuthamaldegurma v.s
    public bool uremeYaparMi();
public booingoyառան()diMi();
    fublic string zehirliMi();
    return "Memeliler";
}
    public virtual string tur()
          return "Yunus";
     public bool uremeYaparMi()
          return true;
     public string zehirliMi()
          return "Zehirsiz";
     }
```

CinNehirYunusu.cs

CinNehirYunusu derives from Yunus and overrides virtual members.

KralKobra.cs

KralKobra implements IHayvan and marks some members virtual.

```
using System;
public chaesfkoelkabaykabaykabayvabanus
    poblice stroting sepolitike politike politike ndiMi()
      h∌ýv¥iumius⊎diaksine cin nehir yun
    publiétkkmikogbépes()i tukenmediginde
      Yneturnamaldegurma v.s
           bueingdemerapagMi())())
     ublic boolngoşünükéndiM
    publicet sitmi higii nië Nejbeni itel Yilliniu su "
                "Memel
                    string tur()
             rtual.
    ∯ublic virtual string tur()
               "Yılan";
        return
                "Yunus"
        return
    bublic bool uremeYaparMi()
    ∮ublic bool uremeYaparMi()
        return true;
        return true;
    public virtual string zehirliMi()
    public string
                  zehirliMi()
        return
        return "Zehirsiz";
```

HintKralKobrasi.cs

HintKralKobrasi derives from KralKobra, overrides behaviour, adds extra method.

```
pshhocSystem; HintKralKobrasi : KralKobra
publicabitinaesvišiasilaidayvitdayvitayvitaidii()
    povibib meto dare metod kao6 o boezsbern Tiike o de Mi√o türet
    poblice stroting soboditke printinkendimi()
    ស្លែ២៤៩៤ volvers adéas strieng clour (e) hir yunusu nes
    publicet ថែលការសិច្ចបទុស្សា tukenmediginden false
    }/ YneturnamHlelegKraa Kobrasi";
    public bueinademeľabamidíú)()
    ∌ublic boolngoyùñùkéndiMi()
    ֆublietunn HMenkKrilKobrasinaOzgunMetod()
    ∌ublic virtual string tur()
    pbublietwimtB@l string tur()
        return "Yilan":
                "Yunus
        return
    ₯ublic bool uremeYaparMi()
```

Istemci.cs

Istemci consumes IHayvan and prints its info, oblivious to concrete type.

Program.cs

Program creates random animal and demonstrates polymorphic behaviour.

```
րոհիցcSystem; HintKralKobrasi : KralKobra
pabepaceaeovine in the property of the propert
                ស្នាប់ត្រង់ខ្លាំប្រជាជាស្រាស់ អាចស្រាស់ ស្រាស់ ស្រាស់ ស្រាស់ អាចស្រាស់ ស្រាស់ អាចស្រាស់ អាចស្រាស់ អាចស្រាស់ អាស្រាស់ អាចស្រាស់ អាចស្រាស
                 poblicz strotobał obrotik opyli Mirkondi Mi()
                 publiconsobbaywantuebishteelebayyanzehrasigeleHayvanVvan(zehigeleBay);
                 អ៊ីប់blioeeurimi ម៉ូច៉ូលែកម៉ូស៊ីម៉ូម៉ាច់ស៊ីស៊ីហ៊ូ)ប្ទេប";
                 խublietaYnondBendaribasKəbnewiManas(տ)Metod()
                 #ublic v@imNehisYunugutainNehirYunusu = new CinNehirYunusu();
                หุ้ublmetwimtนิด) string <mark>tur()</mark>
I retuKmalKobaa"kralKobra = new KralKobra();
                                 retuHintKralKöbrasi hintKralKobrasi = new HintKralKobrasi();
}
                bublic bool uremeYaparMi()
                 public bóólKuarekwewapakMai(Kobra2 = new KralKobra();
                                 return KrueKobra kralKobra3 = new KralKobra();
                                 return true:
                 bublic vIstemciststemcrehimewMIstemci(hintKralKobrasi);
                 ∯ublic string zehirliMi()
                                 feturn "Zehirli";
                                peblin šžebicsIHäyvan rastgeleHayvanUret(int rastgeleSayi)
}
}
                                                 if(rastgeleSayi == 1)
                                                                 Yunus yunus = new Yunus();
                                                                 return yunus;
                                                 else if(rastgeleSayi == 2){
                                                                 CinNehirYunusu cinNehirYunusu = new CinNehirYunusu();
                                                                 return cinNehirYunusu;
                                                 else if(rastgeleSayi == 3){
                                                                KralKobra kralKobra = new KralKobra();
                                                                 return kralKobra;
                                                 else
                                                                 Yunus yunus = new Yunus();
                                                                 return yunus;
             }
}
```

Why Polymorphism Matters

- Write once, work with any derived type.
- Extend the system with new classes without touching client code.
- Keep business logic decoupled from concrete implementations.

Execution Flow

Program selects an IHayvan implementation at runtime and passes it to Istemci.

Istemci calls methods declared on the interface; the CLR dispatches to the correct override based on actual type.