using System;

using System.Collections.Generic;

using System.Text;

namespace InheritanceDemo

{

class Employee

{

int id;

string name;

string department;

public void GetDetails()

{

Console.WriteLine("Enter ID");

id = Convert.ToByte(Console.ReadLine());

Console.WriteLine("Enter Name");

name = Console.ReadLine();

Console.WriteLine("enter Department");

department = Console.ReadLine();

}

public void DisplayDetails()

{

Console.WriteLine("Id is " + id);

Console.WriteLine("Name is " + name);

Console.WriteLine("Department is " + name);

}

}

}

PartTimEmployee

using System;

using System.Collections.Generic;

using System.Text;

namespace InheritanceDemo

{

class Employee

{

int id;

string name;

string department;

public void GetDetails()

{

Console.WriteLine("Enter ID");

id = Convert.ToByte(Console.ReadLine());

Console.WriteLine("Enter Name");

name = Console.ReadLine();

Console.WriteLine("enter Department");

department = Console.ReadLine();

}

public void DisplayDetails()

{

Console.WriteLine("Id is " + id);

Console.WriteLine("Name is " + name);

Console.WriteLine("Department is " + name);

}

}

}

FullTimeEmployee

using System;

using System.Collections.Generic;

using System.Text;

namespace InheritanceDemo

{

class FullTimEmployee : Employee

{

string manager;

int salary;

public void GetFullTimEmployeeDetails()

{

Console.WriteLine("Enter manager");

manager = Console.ReadLine();

Console.WriteLine("enter salary");

salary = Convert.ToInt16(Console.ReadLine());

}

public void DisplayFullTimEmployeeDetails()

{

Console.WriteLine("manager is " + manager);

Console.WriteLine("salary is " + salary);

}

}

}

Program.cs

using System;

namespace InheritanceDemo

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Employee Details");

Employee employee = new Employee();

employee.GetDetails();

employee.DisplayDetails();

Console.WriteLine("Full Employee Details");

FullTimEmployee fullTimEmployee = new FullTimEmployee();

fullTimEmployee.GetDetails();

fullTimEmployee.GetFullTimEmployeeDetails();

fullTimEmployee.DisplayDetails();

fullTimEmployee.DisplayFullTimEmployeeDetails();

Console.WriteLine("PartTime Employee Details");

PartimeEmployee partimeEmployee = new PartimeEmployee();

partimeEmployee.GetDetails();

partimeEmployee.GetPartimeEmployeeDetails();

partimeEmployee.DisplayDetails();

partimeEmployee.DisplayPartimeEmployeeDetails();

}

}

}

Method Overriding

using System;

using System.Collections.Generic;

using System.Text;

namespace InheritanceDemo

{

class PartimeEmployee : Employee

{

int duration;

string projectName;

public void GetDetails() // same as the parent class

{

base.GetDetails(); // this will call GetDetails() method of parent class

Console.WriteLine("Enter projectName");

projectName = Console.ReadLine();

Console.WriteLine("enter duration");

duration = Convert.ToInt16(Console.ReadLine());

}

public void DisplayDetails()

{

base.DisplayDetails();

Console.WriteLine("projectName is " + projectName);

Console.WriteLine("duration is " + duration);

}

}

}

Polymorphism cud be of 2 types

1. Compile Time
2. Run Time

Compile Time Polymorphism is also kn as Static linking Or Early Linking

Is Achieved by using Method Overloading

Run Time : Its uses late or dynamic linking

Is Achieved by using Method Overriding & Virtual Functions