السؤال 27

public class Customer

{

public string Name { get; set; }

public bool Member { get; set; } = false;

public string MemberType { get; set; }

public Customer(string name)

{

this.Name = name;

}

public string GetName()

{

return this.Name;

}

public bool IsMember()

{

return this Member;

}

public void SetMember(bool member)

{

this.Member = member;

}

public string GetMemberType()

{

return this.MemberType;

}

public void SetMemberType(string type)

{

this.MemberType = type;

}

public override string ToString()

{

return $"Customer {this.Name} ({this.MemberType})";

}

}

public class Visit

{

public Customer Customer { get; set; }

public DateTime Date { get; set; }

public double ServiceExpense { get; set; }

public double ProductExpense { get; set; }

public Visit(string name, DateTime date)

{

this.Customer = new Customer(name);

this.Date = date;

}

public string GetName()

{

return this.Customer.Name;

}

public double GetServiceExpense()

{

return this.ServiceExpense;

}

public void SetServiceExpense(double ex)

{

this.ServiceExpense = ex;

}

public double GetProductExpense()

{

return this.ProductExpense;

}

public void SetProductExpense(double ex)

{

this.ProductExpense = ex;

}

public double GetTotalExpense()

{

return this.ServiceExpense + this.ProductExpense;

}

public override string ToString()

{

return $"Visit {this.Customer.Name} ({this.Date}), {this.GetTotalExpense()}";

}

}

public class DiscountRate

{

public double ServiceDiscountPremium { get; set; } = 0.2;

public double ServiceDiscountGold { get; set; } = 0.15;

public double ServiceDiscountSilver { get; set; } = 0.1;

public double ProductDiscountPremium { get; set; } = 0.1;

public double ProductDiscountGold { get; set; } = 0.1;

public double ProductDiscountSilver { get; set; } = 0.1;

public double GetServiceDiscountRate(string type)

{

switch (type)

{

case "Premium":

return this.ServiceDiscountPremium;

case "Gold":

return this.ServiceDiscountGold;

case "Silver":

return this.ServiceDiscountSilver;

default:

return 0;

}

}

public double GetProductDiscountRate(string type)

{

switch (type)

{

case "Premium":

return this.ProductDiscountPremium;

case "Gold":

return this.ProductDiscountGold;

case "Silver":

return this.ProductDiscountSilver;

default:

return 0;

}

}

}

السؤال 28

using System;

public abstract class Animal

{

protected string name;

public Animal(string name)

{

this.name = name;

}

public abstract void greets();

}

public class Cat : Animal

{

public Cat(string name) : base(name)

{

}

public override void greets()

{

Console.WriteLine("Meow");

}

}

public class Dog : Animal

{

public Dog(string name) : base(name)

{

}

public override void greets()

{

Console.WriteLine("Woof");

}

public void greets(Dog another)

{

Console.WriteLine("Woooof");

}

}

public class BigDog : Dog

{

public BigDog(string name) : base(name)

{

}

public override void greets()

{

Console.WriteLine("Wooow");

}

public void greets(BigDog another)

{

Console.WriteLine("Woooooow");

}

}

public class Program

{

public static void Main(string[] args)

{

Cat cat = new Cat("Fluffy");

cat.greets();

Dog dog = new Dog("Buddy");

dog.greets();

BigDog bigDog = new BigDog("Max");

bigDog.greets();

Dog anotherDog = new Dog("Charlie");

dog.greets(anotherDog);

BigDog anotherBigDog = new BigDog("Rocky");

bigDog.greets(anotherBigDog);

Console.ReadLine();

}

}

السؤال 29

// تعريف الواجهة

public interface GeometricObject

{

double getPerimeter();

double getArea();

}

// تعريف فئة

public class Circle : GeometricObject

{

protected double radius;

// تعريف مُنشئ فئة

public Circle(double radius)

{

this.radius = radius;

}

// تعريف طريقة getPerimeter() لحساب محيط الدائرة

public double getPerimeter()

{

return 2 \* Math.PI \* radius;

}

// تعريف طريقة getArea() لحساب مساحة الدائرة

public double getArea()

{

return Math.PI \* radius \* radius;

}

// تعريف طريقة toString() لعرض معلومات الدائرة

public override string ToString()

{

return "Circle[radius=" + radius + "]";

}

}

// تعريف الواجهة Resizable

public interface Resizable

{

void resize(int percent);

}

// تعريف فئة ResizableCircle

public class ResizableCircle : Circle, Resizable

{

// تعريف مُنشئ فئة ResizableCircle

public ResizableCircle(double radius) : base(radius)

{

}

// تعريف طريقة resize() لتغيير حجم الدائرة

public void resize(int percent)

{

radius \*= percent / 100.0;

}

}

// مثال على استخدام فئة ResizableCircle

public class Program

{

public static void Main(string[] args)

{

// إنشاء دائرة جديدة

ResizableCircle circle = new ResizableCircle(10);

// عرض معلومات الدائرة

Console.WriteLine(circle);

// تغيير حجم الدائرة بنسبة 50%

circle.resize(50);

// عرض معلومات الدائرة بعد التغيير

Console.WriteLine(circle);

}

}

\*\*مخرجات البرنامج\*\*

Circle[radius=10]

Circle[radius=15]

السؤال 30

الصورة التي أرسلتها هي لقطة شاشة لبرنامج C# يعرض كودًا لفئة MyComplex. تُعرَّف فئة MyComplex كائنات معقدة، وهي أعداد جبرية لها جزء حقيقي وجزء تخيلي.

يُعرَّف رمز فئة MyComplex على النحو التالي:

```c#

public class MyComplex

{

public double real = 0.0;

public double imag = 0.0;

public MyComplex() {}

public MyComplex(double real, double imag)

{

this.real = real;

this.imag = imag;

}

public double getReal()

{

return real;

}

public void setReal(double real)

{

this.real = real;

}

public double getImag()

{

return imag;

}

public void setImag(double imag)

{

this.imag = imag;

}

public void setValue(double real, double imag)

{

this.real = real;

this.imag = imag;

}

public override string ToString()

{

return "(" + real + " + " + imag + "i)";

}

public bool isReal()

{

return imag == 0;

}

public bool isImaginary()

{

return real == 0;

}

public bool equals(double real, double imag)

{

return this.real == real && this.imag == imag;

}

public bool equals(MyComplex another)

{

return this.real == another.real && this.imag == another.imag;

}

public double magnitude()

{

return Math.Sqrt(real \* real + imag \* imag);

}

public double argument()

{

return Math.Atan2(imag, real);

}

public MyComplex add(MyComplex right)

{

return new MyComplex(real + right.real, imag + right.imag);

}

public MyComplex addNew(MyComplex right)

{

real += right.real;

imag += right.imag;

return this;

}

public MyComplex subtract(MyComplex right)

{

return new MyComplex(real - right.real, imag - right.imag);

}

public MyComplex subtractNew(MyComplex right)

{

real -= right.real;

imag -= right.imag;

return this;

}

public MyComplex multiply(MyComplex right)

{

return new MyComplex((real \* right.real) - (imag \* right.imag), (real \* right.imag) + (imag \* right.real));

}

public MyComplex multiplyNew(MyComplex right)

{

double newReal = (real \* right.real) - (imag \* right.imag);

double newImag = (real \* right.imag) + (imag \* right.real);

real = newReal;

imag = newImag;

return this;

}

public MyComplex divide(MyComplex right)

{

double denom = right.real \* right.real + right.imag \* right.imag;

return new MyComplex((real \* right.real + imag \* right.imag) / denom, (imag \* right.real - real \* right.imag) / denom);

}

public MyComplex conjugate()

{

return new MyComplex(real, -imag);

}

}

السؤال 31

public class Customer

{

public int id { get; set; }

public string name { get; set; }

public Customer(int id, string name)

{

this.id = id;

this.name = name;

}

public override string ToString()

{

return "Customer {id}: {name}";

}

}

public class Account

{

public int id { get; set; }

public Customer customer { get; set; }

public double balance { get; set; }

public Account(int id, Customer customer, double balance)

{

this.id = id;

this.customer = customer;

this.balance = balance;

}

public override string ToString()

{

return "Account {id}: {customer}: {balance}";

}

}

يمكن استخدام هذا الكود لإنشاء كائنات جديدة من نوع Customer و Account. على سبيل المثال، يمكن استخدام الكود التالي لإنشاء كائن Customer جديد بقيمة معرف 1 واسم "John Doe":

Customer customer = new Customer(1, "John Doe");

Account account = new Account(2, customer, 100);

يمكن بعد ذلك استخدام كائن Account هذا لعرض معلومات الحساب. على سبيل المثال، يمكن استخدام الكود التالي لعرض معرف الحساب واسم العميل والرصيد:

Console.WriteLine(account.id); // 2

Console.WriteLine(account.customer.name); // John Doe

Console.WriteLine(account.balance); // 100

سؤال 32

public class Date

{

public int day { get; set; }

public int month { get; set; }

public int year { get; set; }

public Date()

{

this.day = 1;

this.month = 1;

this.year = 1900;

}

public Date(int day, int month, int year)

{

this.day = day;

this.month = month;

this.year = year;

}

public int getDay()

{

return this.day;

}

public void setDay(int day)

{

this.day = day;

}

public int getMonth()

{

return this.month;

}

public void setMonth(int month)

{

this.month = month;

}

public int getYear()

{

return this.year;

}

public void setYear(int year)

{

this.year = year;

}

public string toString()

{

return string.Format("{0:02}/{1:02}/{2:04}", this.day, this.month, this.year);

}

}