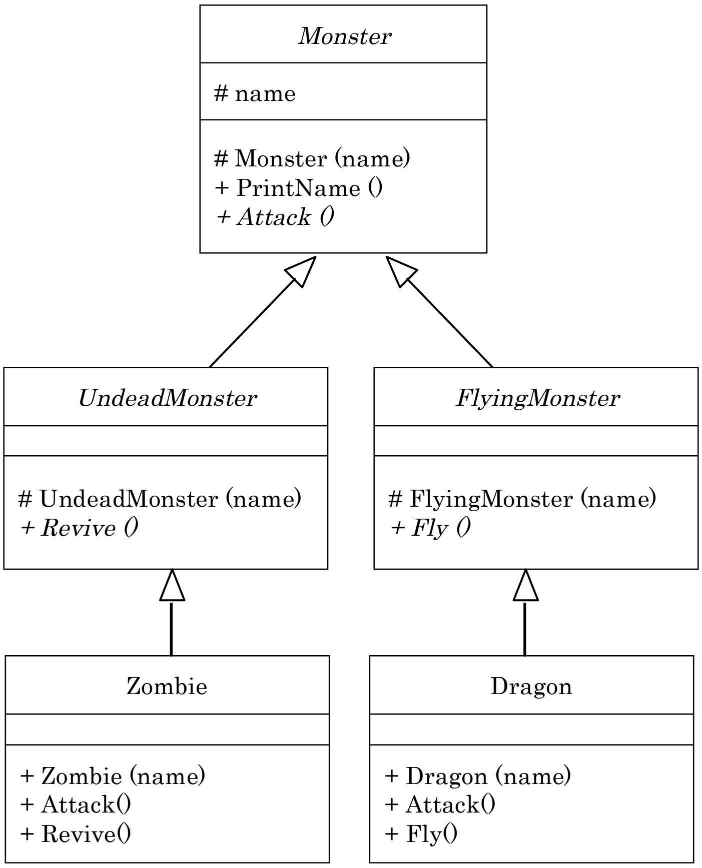
オブジェクト指向プログラミング　第１０回演習　　　年　　組　　番 氏名

1.　右の UML のクラス図で示したクラスを作成し，下記のプログラムを完成させて動作を理解せよ。




```
class Program {
  public static void Main() {
    Zombie z1 = new Zombie("ゾンビ１号");
    z1.PrintName();
    z1.Attack();
    z1.Revive();

    Dragon d1 = new Dragon("ドラゴン");
    d1.PrintName();
    d1.Attack();
    d1.Fly();
    Console.WriteLine();

    // 基本クラスの変数への代入と多態性の利用
    Monster m1 = new Zombie("ゾンビ２号");
    m1.Attack();
    Monster m2;
    m2 = d1;
    m2.Attack();
    Console.WriteLine();

    // 実行時型判定 (is 演算子)
    Console.WriteLine(m1 is Zombie);
    Console.WriteLine(m1 is UndeadMonster);
    Console.WriteLine(m2 is Zombie);

    // 派生クラスから基本クラスへの型変換
    //（as 演算子による“ダウンキャスト”）
    Zombie z = m1 as Zombie;
    z.Revive();

    // 基本クラスの仮引数としてメソッドに渡す
    AttackTwice(z1);
    AttackTwice(d1);
    Console.WriteLine();

    Monster [] a = new Monster[4];
    a[0] = z1;  a[1] = d1;  a[2] = m1;  a[3] = m2;

    foreach (Monster m in a) {
      m.PrintName();
      // is 演算子で確認後，通常のキャストで変換
      if (m is UndeadMonster) {
        UndeadMonster u = (UndeadMonster) m;
        u.Revive();
      }
      // as 演算子でキャスト後，変換できたか判定
      FlyingMonster f = m as FlyingMonster;
      if (f != null) f.Fly();
    }
  }

  public static void AttackTwice(Monster m) {
    m.Attack();
    m.Attack();
  }
}
```