



Zaccaria El Kacimi

1WA

Block 4

15.04.24 – 18.04.24

1) Bücher

Buch Klasse

```
6 Verweise
class Buch
{
    private string Title;
    private string Author;
    private int ISBN;
    private string Genre;
    private string Verlag;
    private DateTime Erscheinungsjahr;
    private string Kommentar;

    1 Verweis
    public Buch()
    {
        Title = "";
        Author = "";
        ISBN = 0;
        Genre = "";
        Verlag = "";
        Erscheinungsjahr = DateTime.MinValue;
        Kommentar = "";
    }

    0 Verweise
    public Buch(string title, string author)
    {
        Title = title;
        Author = author;
        ISBN = 0;
        Genre = "";
        Verlag = "";
        Erscheinungsjahr = DateTime.MinValue;
        Kommentar = "";
    }

    0 Verweise
    public Buch(string title, string author, int isbn, DateTime erscheinungsjahr)
    {
        Title = title;
        Author = author;
        ISBN = isbn;
        Genre = "";
        Verlag = "";
        Erscheinungsjahr = erscheinungsjahr;
        Kommentar = "";
    }

    0 Verweise
    public Buch(string title, string author, int isbn, string genre, string verlag, DateTime erscheinungsjahr, string kommentar)
    {
        Title = title;
        Author = author;
        ISBN = isbn;
        Genre = genre;
        Verlag = verlag;
        Erscheinungsjahr = erscheinungsjahr;
        Kommentar = kommentar;
    }

    //get , set Title
    1 Verweis
    public string get_Title() { return Title; }
    1 Verweis
    public void set_Title(string title) { Title = title; }

    //get , set Author
    0 Verweise
    public string get_Author() { return Author; }
    0 Verweise
    public void set_Author(string author) { Author = author; }

    //get , set ISBN
    0 Verweise
    public int get_ISBN() { return ISBN; }
    0 Verweise
    public void set_ISBN(int isbn) { ISBN = isbn; }

    //get , set Genre
    0 Verweise
    public string get_Genre() { return Genre; }
    0 Verweise
    public void set_Genre(string genre) { Genre = genre; }

    //get , set Verlags
    0 Verweise
    public string get_Verlag() { return Genre; }
    0 Verweise
    public void set_Verlag(string verlag) { Verlag = verlag; }

    //get , set Erscheinungsjahr
    0 Verweise
    public DateTime get_Erscheinungsjahr() { return Erscheinungsjahr; }
    0 Verweise
    public void set_Erscheinungsjahr(DateTime erscheinungsjahr) { Erscheinungsjahr = erscheinungsjahr; }

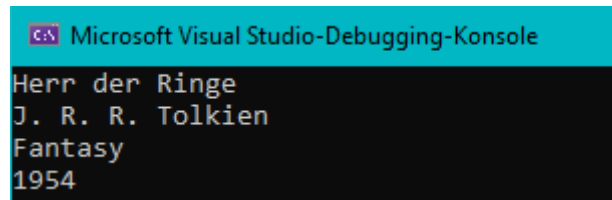
    //get , set Kommentar
    0 Verweise
    public string get_Kommentar() { return Kommentar; }
    0 Verweise
    public void set_Kommentar(string kommentar) { Kommentar = kommentar; }
}
```

Main

```
0 Verweise
static void Main(string[] args)
{
    //test example
    Buch b = new Buch();
    b.set_Title("Herr der Ringe");
    b.set_Author("J. R. R. Tolkien");
    b.set_Genre("Fantasy");
    b.set_Erscheinungsjahr(1954);

    Console.WriteLine($"{b.get_Title()}");
    Console.WriteLine($"{b.get_Author()}");
    Console.WriteLine($"{b.get_Genre()}");
    Console.WriteLine($"{b.get_Erscheinungsjahr()}");
}
```

Konsole



The screenshot shows the output of the program in the Visual Studio console. The title bar reads "Microsoft Visual Studio-Debugging-Konsole". The output consists of four lines: "Herr der Ringe", "J. R. R. Tolkien", "Fantasy", and "1954".

```
Microsoft Visual Studio-Debugging-Konsole
Herr der Ringe
J. R. R. Tolkien
Fantasy
1954
```

2) Schauspieler

Schauspieler Klasse

```
6 Verweise
class Schauspieler
{
    private string Vorname;
    private string Nachname;
    private string Spitzname;
    private string Geburtsort;
    private string Geburtsdatum;
    private string Nationalitaet;
    private double Groesse;

    1 Verweis
    public Schauspieler()
    {
        Vorname = "";
        Nachname = "";
        Spitzname = "";
        Geburtsort = "";
        Geburtsdatum = "";
        Nationalitaet = "";
        Groesse = 0;
    }

    0 Verweise
    public Schauspieler(string vorname, string nachname)
    {
        Vorname = vorname;
        Nachname = nachname;
        Spitzname = "";
        Geburtsort = "";
        Geburtsdatum = "";
        Nationalitaet = "";
        Groesse = 0;
    }

    0 Verweise
    public Schauspieler(string vorname, string nachname, string spitzname, double groesse)
    {
        Vorname = vorname;
        Nachname = nachname;
        Spitzname = spitzname;
        Geburtsort = "";
        Geburtsdatum = "";
        Nationalitaet = "";
        Groesse = groesse;
    }

    0 Verweise
    public Schauspieler(string vorname, string nachname, string spitzname, string geburtsort, string geburtsdatum, string nationalitaet, double groesse)
    {
        Vorname = vorname;
        Nachname = nachname;
        Spitzname = spitzname;
        Geburtsort = geburtsort;
        Geburtsdatum = geburtsdatum;
        Nationalitaet = nationalitaet;
        Groesse = groesse;
    }

    //get , set Vorname
    1 Verweis
    public string get_Vorname() { return Vorname; }
    1 Verweis
    public void set_Vorname(string vorname) { Vorname = vorname; }

    //get , set Nachname
    1 Verweis
    public string get_Nachname() { return Nachname; }
    1 Verweis
    public void set_Nachname(string nachname) { Nachname = nachname; }

    //get , set Spitzname
    0 Verweise
    public string get_Spitzname() { return Spitzname; }
    0 Verweise
    public void set_Spitzname(string spitzname) { Spitzname = spitzname; }

    //get , set Geburtsort
    1 Verweis
    public string get_Geburtsort() { return Geburtsort; }
    1 Verweis
    public void set_Geburtsort(string geburtsort) { Geburtsort = geburtsort; }

    //get , set Geburtsdatum
    1 Verweis
    public string get_Geburtsdatum() { return Geburtsdatum; }
    1 Verweis
    public void set_Geburtsdatum(string geburtsdatum) { Geburtsdatum = geburtsdatum; }

    //get , set Nationalitaet
    1 Verweis
    public string get_Nationalitaet() { return Nationalitaet; }
    1 Verweis
    public void set_Nationalitaet(string nationalitaet) { Nationalitaet = nationalitaet; }

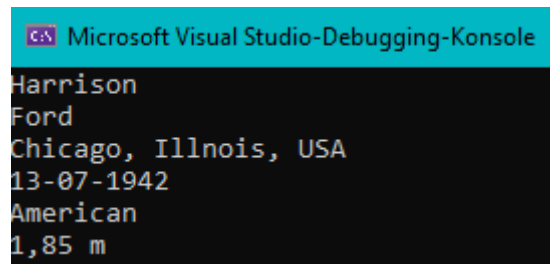
    //get , set Groesse
    1 Verweis
    public double get_Groesse() { return Groesse; }
    1 Verweis
    public void set_Groesse(double groesse) { Groesse = groesse; }
}
```

Main

```
0 Verweise
static void Main(string[] args)
{
    //test example
    Schauspieler b = new Schauspieler();
    b.set_Vorname("Harrison");
    b.set_Nachname("Ford");
    b.set_Geburtsort("Chicago, Illinois, USA");
    b.set_Geburtsdatum("13-07-1942");
    b.set_Nationalitaet("American");
    b.set_Groesse(1.85);

    Console.WriteLine($"{b.get_Vorname()}");
    Console.WriteLine($"{b.get_Nachname()}");
    Console.WriteLine($"{b.get_Geburtsort()}");
    Console.WriteLine($"{b.get_Geburtsdatum()}");
    Console.WriteLine($"{b.get_Nationalitaet()}");
    Console.WriteLine($"{b.get_Groesse()} m");
}
```

Konsole



Microsoft Visual Studio-Debugging-Konsole

```
Harrison
Ford
Chicago, Illinois, USA
13-07-1942
American
1,85 m
```

3) Filme

Film Klasse

```
6 Verweise
class Film
{
    private string Title;
    private int Erscheinungsjahr;
    private string Genre;
    private double Laufzeit;
    private string[] Cast;
    private string Sprache;
    private int FSK;

    1 Verweis
    public Film()
    {
        Title = "";
        Erscheinungsjahr = 0;
        Genre = "";
        Laufzeit = 0;
        Cast = [];
        Sprache = "";
        FSK = 0;
    }

    0 Verweise
    public Film(string title, int erscheinungsjahr)
    {
        Title = title;
        Erscheinungsjahr = erscheinungsjahr;
        Genre = "";
        Laufzeit = 0;
        Cast = [];
        Sprache = "";
        FSK = 0;
    }

    0 Verweise
    public Film(string title, int erscheinungsjahr, string genre, double laufzeit)
    {
        Title = title;
        Erscheinungsjahr = erscheinungsjahr;
        Genre = genre;
        Laufzeit = laufzeit;
        Cast = [];
        Sprache = "";
        FSK = 0;
    }

    0 Verweise
    public Film(string title, int erscheinungsjahr, string genre, double laufzeit, string cast, string sprache, int fsk)
    {
        Title = title;
        Erscheinungsjahr = erscheinungsjahr;
        Genre = genre;
        Laufzeit = laufzeit;
        Cast = cast.Split(',');
        Sprache = sprache;
        FSK = fsk;
    }

    //get , set Title
    1 Verweis
    public string get_Title() { return Title; }
    1 Verweis
    public void set_Title(string title) { Title = title; }

    //get , set Erscheinungsjahr
    1 Verweis
    public int get_Erscheinungsjahr() { return Erscheinungsjahr; }
    1 Verweis
    public void set_Erscheinungsjahr(int erscheinungsjahr) { Erscheinungsjahr = erscheinungsjahr; }

    //get , set Genre
    1 Verweis
    public string get_Genre() { return Genre; }
    1 Verweis
    public void set_Genre(string genre) { Genre = genre; }

    //get , set Laufzeit
    1 Verweis
    public double get_Laufzeit() { return Laufzeit; }
    1 Verweis
    public void set_Laufzeit(int laufzeit)
    {
        double hours = Convert.ToDouble($"{laufzeit / 60},{laufzeit % 60}");
        Laufzeit = (laufzeit > 60) ? hours : laufzeit;
    }

    //get , set Cast
    1 Verweis
    public string get_Cast()
    {
        string castlist = "";
        if (Cast.Length > 0)
        {
            for (int i = 0; i < Cast.Length; i++)
            {
                castlist += $"{Cast[i].ToString()} ";
            }
        }
        return castlist;
    }
    1 Verweis
    public void set_Cast(string cast) { Cast = cast.Split(','); }

    //get , set Sprache
    0 Verweise
    public string get_Sprache() { return Sprache; }
    0 Verweise
    public void set_Sprache(string sprache) { Sprache = sprache; }

    //get , set FSK
    0 Verweise
    public int get_FSK() { return FSK; }
    0 Verweise
    public void set_FSK(int fsk) { FSK = fsk; }
}
```

Main

0 Verweise

```
static void Main(string[] args)
{
    //test example
    Film f = new Film();
    f.set_Title("Bube Dame König grAS");
    f.set_Erscheinungsjahr(1998);
    f.set_Genre("Crime");
    f.set_Laufzeit(107);
    f.set_Cast("Jason Flemyng,Jason Statham,Nick Moran,Dexter Fletcher,Steven Mackintosh,Vinnie Jones");

    Console.WriteLine($"{f.get_Title()}");
    Console.WriteLine($"{f.get_Erscheinungsjahr()}");
    Console.WriteLine($"{f.get_Genre()}");
    Console.WriteLine($"{f.get_Laufzeit()}");
    Console.WriteLine($"{f.get_Cast()}");
}
```

Konsole



Microsoft Visual Studio-Debugging-Konsole

```
Bube Dame König grAS
1998
Crime
1,47

Jason Flemyng
Jason Statham
Nick Moran
Dexter Fletcher
Steven Mackintosh
Vinnie Jones
```

4) Clash of Clans

Held Klasse

```
6 Verweise
class Held
{
    private string Name;
    private int Level;
    private int Kosten;
    private int Trefferpunkte;
    private int Regenerationszeit;
    private int Schaden;
    private string Spezialfähigkeit;

    1 Verweis
    public Held()
    {
        Name = "";
        Level = 0;
        Kosten = 0;
        Trefferpunkte = 0;
        Regenerationszeit = 0;
        Schaden = 0;
        Spezialfähigkeit = "";
    }

    0 Verweise
    public Held(string name, int level)
    {
        Name = name;
        Level = level;
        Kosten = 0;
        Trefferpunkte = 0;
        Regenerationszeit = 0;
        Schaden = 0;
        Spezialfähigkeit = "";
    }

    0 Verweise
    public Held(string name, int level, int kosten, int trefferpunkte)
    {
        Name = name;
        Level = level;
        Kosten = kosten;
        Trefferpunkte = trefferpunkte;
        Regenerationszeit = 0;
        Schaden = 0;
        Spezialfähigkeit = "";
    }

    0 Verweise
    public Held(string name, int level, int kosten, int trefferpunkte, int regenerationszeit, int schaden, string spezialfähigkeit)
    {
        Name = name;
        Level = level;
        Kosten = kosten;
        Trefferpunkte = trefferpunkte;
        Regenerationszeit = regenerationszeit;
        Schaden = schaden;
        Spezialfähigkeit = spezialfähigkeit;
    }

    //get , set Name
    1 Verweis
    public string get_Name() { return Name; }
    1 Verweis
    public void set_Name(string name) { Name = name; }

    //get , set Level
    1 Verweis
    public int get_Level() { return Level; }
    1 Verweis
    public void set_Level(int level) { Level = level; }

    //get , set Kosten
    1 Verweis
    public int get_Kosten() { return Kosten; }
    1 Verweis
    public void set_Kosten(int kosten) { Kosten = kosten; }

    //get , set Trefferpunkte
    1 Verweis
    public int get_Trefferpunkte() { return Trefferpunkte; }
    1 Verweis
    public void set_Trefferpunkte(int trefferpunkte) { Trefferpunkte = trefferpunkte; }

    //get , set Regenerationszeit
    0 Verweise
    public int get_Regenerationszeit() { return Regenerationszeit; }
    0 Verweise
    public void set_Regenerationszeit(int regenerationszeit) { Regenerationszeit = regenerationszeit; }

    //get , set Schaden
    1 Verweis
    public int get_Schaden() { return Schaden; }
    1 Verweis
    public void set_Schaden(int schaden) { Schaden = schaden; }

    //get , set Level
    0 Verweise
    public string get_Spezialfähigkeit() { return Spezialfähigkeit; }
    0 Verweise
    public void set_Spezialfähigkeit(string spezialfähigkeit) { Spezialfähigkeit = spezialfähigkeit; }
}
```


Trupp Klasse

```
1 Vorwissen
public Trupp()
{
    Name = "";
    Level = 0;
    Entwicklungskosten = 0;
    Baukosten = 0;
    Elixier = 0;
    Trefferpunkte = 0;
    Schaden = 0;
    Bewegungsgeschwindigkeit = 0;
    Angriffsgeschwindigkeit = 0;
    Reichweite = 0;
    Wohnraum = 0;
    Einzel = 0;
}

0 Vorwissen
public Trupp(string name, int level, int entwicklungskosten, int baukosten)
{
    Name = name;
    Level = level;
    Entwicklungskosten = entwicklungskosten;
    Baukosten = baukosten;
    Elixier = 0;
    Trefferpunkte = 0;
    Schaden = 0;
    Bewegungsgeschwindigkeit = 0;
    Angriffsgeschwindigkeit = 0;
    Reichweite = 0;
    Wohnraum = 0;
    Einzel = 0;
}

0 Vorwissen
public Trupp(string name, int level, int entwicklungskosten, int baukosten, int elixier, int reichweite)
{
    Name = name;
    Level = level;
    Entwicklungskosten = entwicklungskosten;
    Baukosten = baukosten;
    Elixier = elixier;
    Trefferpunkte = 0;
    Schaden = 0;
    Bewegungsgeschwindigkeit = 0;
    Angriffsgeschwindigkeit = 0;
    Reichweite = reichweite;
    Wohnraum = 0;
    Einzel = 0;
}

0 Vorwissen
public Trupp(string name, int level, int entwicklungskosten, int baukosten, int elixier, int trefferpunkte, int schaden,
    int bewegungsgeschwindigkeit, int angriffsgeschwindigkeit, int reichweite, int wohnraum, int einzel)
{
    Name = name;
    Level = level;
    Entwicklungskosten = entwicklungskosten;
    Baukosten = baukosten;
    Elixier = elixier;
    Trefferpunkte = trefferpunkte;
    Schaden = schaden;
    Bewegungsgeschwindigkeit = bewegungsgeschwindigkeit;
    Angriffsgeschwindigkeit = angriffsgeschwindigkeit;
    Reichweite = reichweite;
    Wohnraum = wohnraum;
    Einzel = einzel;
}

//get , set Name
1 Vorwissen
public string get_Name() { return Name; }
1 Vorwissen
public void set_Name(string name) { Name = name; }

//get , set Level
1 Vorwissen
public int get_Level() { return Level; }
1 Vorwissen
public void set_Level(int level) { Level = level; }

//get , set Entwicklungskosten
1 Vorwissen
public int get_Entwicklungskosten() { return Entwicklungskosten; }
1 Vorwissen
public void set_Entwicklungskosten(int entwicklungskosten) { Entwicklungskosten = entwicklungskosten; }

//get , set Baukosten
0 Vorwissen
public int get_Baukosten() { return Baukosten; }
0 Vorwissen
public void set_Baukosten(int baukosten) { Baukosten = baukosten; }

//get , set Elixier
0 Vorwissen
public int get_Elixier() { return Elixier; }
0 Vorwissen
public void set_Elixier(int elixier) { Elixier = elixier; }

//get , set Trefferpunkte
1 Vorwissen
public int get_Trefferpunkte() { return Trefferpunkte; }
1 Vorwissen
public void set_Trefferpunkte(int trefferpunkte) { Trefferpunkte = trefferpunkte; }

//get , set Schaden
1 Vorwissen
public int get_Schaden() { return Schaden; }
1 Vorwissen
public void set_Schaden(int schaden) { Schaden = schaden; }

//get , set Bewegungsgeschwindigkeit
0 Vorwissen
public int get_Bewegungsgeschwindigkeit() { return Bewegungsgeschwindigkeit; }
0 Vorwissen
public void set_Bewegungsgeschwindigkeit(int bewegungsgeschwindigkeit) { Bewegungsgeschwindigkeit = bewegungsgeschwindigkeit; }

//get , set Angriffsgeschwindigkeit
0 Vorwissen
public int get_Angriffsgeschwindigkeit() { return Angriffsgeschwindigkeit; }
0 Vorwissen
public void set_Angriffsgeschwindigkeit(int angriffsgeschwindigkeit) { Angriffsgeschwindigkeit = angriffsgeschwindigkeit; }

//get , set Reichweite
0 Vorwissen
public int get_Reichweite() { return Reichweite; }
0 Vorwissen
public void set_Reichweite(int reichweite) { Reichweite = reichweite; }

//get , set Wohnraum
0 Vorwissen
public int get_Wohnraum() { return Wohnraum; }
0 Vorwissen
public void set_Wohnraum(int wohnraum) { Wohnraum = wohnraum; }

//get , set Einzel
0 Vorwissen
public int get_Einzel() { return Einzel; }
0 Vorwissen
public void set_Einzel(int einzel) { Einzel = einzel; }
```

Main

```
0 Vorweise
static void Main(string[] args)
{
    //test example
    Held h = new Held();
    h.set_Name("Riesen");
    h.set_Level(5);
    h.set_Schaden(31);
    h.set_Trefferpunkte(670);
    h.set_Kosten(2250);

    Console.WriteLine($"name: {h.get_Name()}");
    Console.WriteLine($"level: {h.get_Level()}");
    Console.WriteLine($"schaden: {h.get_Schaden()}");
    Console.WriteLine($"treffpunkte: {h.get_Trefferpunkte()}");
    Console.WriteLine($"kosten: {h.get_Kosten()}");

    Console.WriteLine();
    Trupp t = new Trupp();
    t.set_Name("Barbar");
    t.set_Level(4);
    t.set_Entwicklungskosten(200000);
    t.set_Schaden(18);
    t.set_Trefferpunkte(85);

    Console.WriteLine($"name: {t.get_Name()}");
    Console.WriteLine($"level: {t.get_Level()}");
    Console.WriteLine($"entwicklung kosten: {t.get_Entwicklungskosten()}");
    Console.WriteLine($"schaden: {t.get_Schaden()}");
    Console.WriteLine($"treffpunkte: {t.get_Trefferpunkte()}");
}
```

Konsole



The screenshot shows the output of the Main method in the Visual Studio console. The output is as follows:

```
name: Riesen
level: 5
schaden: 31
treffpunkte: 670
kosten: 2250

name: Barbar
level: 4
entwicklung kosten: 200000
schaden: 18
treffpunkte: 85
```

5) Taschenrechner

Code

```
2 Verweise
class Taschenrechner
{
    1 Verweis
    public int Addition(int x, int y) { return x + y; }
    1 Verweis
    public int Subtraction(int x, int y) { return x - y; }
    1 Verweis
    public int Multiplikation(int x, int y) { return x * y; }
    1 Verweis
    public int Division(int x, int y) { return x / y; }

    1 Verweis
    public double Potenzieren(int x, int y) { return Math.Pow(x, y); }
    1 Verweis
    public double WurzelZiehen(int x) { return Math.Sqrt(x); }
    1 Verweis
    public double Logarithmus(int x) { return Math.Log(x); }

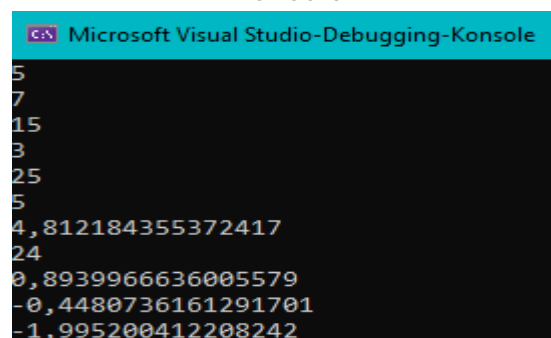
    1 Verweis
    public int Fakultaet(int x)
    {
        int result = 1;
        for (int i = x; i > 0; i--) { result *= i; }
        return result;
    }

    1 Verweis
    public double Sinus(int x) { return Math.Sin(x); }
    1 Verweis
    public double Cosinus(int x) { return Math.Cos(x); }
    1 Verweis
    public double Tangens(int x) { return Math.Tan(x); }
}

0 Verweise
class Program
{
    0 Verweise
    static void Main(string[] args)
    {
        //test example
        Taschenrechner t = new Taschenrechner();

        Console.WriteLine(t.Addition(2,3)); //5
        Console.WriteLine(t.Subtraction(10, 3)); //7
        Console.WriteLine(t.Multiplikation(3, 5)); //15
        Console.WriteLine(t.Division(6, 2)); //3
        Console.WriteLine(t.Potenzieren(5, 2)); //25
        Console.WriteLine(t.WurzelZiehen(25)); //5
        Console.WriteLine(t.Logarithmus(123)); //4.812184355...
        Console.WriteLine(t.Fakultaet(4)); //24
        Console.WriteLine(t.Sinus(90)); //0.89399666...
        Console.WriteLine(t.Cosinus(90)); //-0.44807361...
        Console.WriteLine(t.Tangens(90)); //-1.99530041...
    }
}
```

Konsole



```
Microsoft Visual Studio-Debugging-Konsole
5
7
15
3
25
5
4,812184355372417
24
0,8939966636005579
-0,4480736161291701
-1,995200412208242
```

6) Bruchrechnen

Bruchrechner Klasse

```
2 Versions
class Taschenrechner
{
    1 Versions
    public string Addition(int a1, int a2, int b1, int b2) // a1/a2 + b1/b2
    {
        int numerator = (a1 * b2 + a2 * b1);
        int denominator = (a2 * b2);

        string result = "";

        if (numerator > denominator || numerator < denominator)
        {
            if (numerator % denominator != 0)
            {
                result = $"{numerator / denominator} {numerator % denominator}/{denominator}";
            }
            else
            {
                result = $"{numerator / denominator}";
            }
        }
        else if (denominator == numerator)
        {
            result = "1";
        }
        return result;
    }

    1 Versions
    public string Subtraction(int a1, int a2, int b1, int b2) // a1/a2 - b1/b2
    {
        int numerator = (a1 * b2 - a2 * b1);
        int denominator = (a2 * b2);

        string result = "";

        if (numerator > denominator || numerator < denominator)
        {
            if (numerator % denominator != 0)
            {
                result = $"{numerator / denominator} {numerator % denominator}/{denominator}";
            }
            else
            {
                result = $"{numerator / denominator}";
            }
        }
        else if (denominator == numerator)
        {
            result = "1";
        }
        return result;
    }

    1 Versions
    public string Multiplikation(int a1, int a2, int b1, int b2) // a1/a2 * b1/b2
    {
        int numerator = (a1 * b1);
        int denominator = (a2 * b2);

        string result = "";

        if (numerator > denominator || numerator < denominator)
        {
            if (numerator % denominator != 0)
            {
                result = $"{numerator / denominator} {numerator % denominator}/{denominator}";
            }
            else
            {
                result = $"{numerator / denominator}";
            }
        }
        else if (denominator == numerator)
        {
            result = "1";
        }
        return result;
    }

    1 Versions
    public string Division(int a1, int a2, int b1, int b2) // a1/a2 : b1/b2
    {
        int numerator = (a1 * b2);
        int denominator = (a2 * b1);

        string result = "";

        if (numerator > denominator || numerator < denominator)
        {
            if (numerator % denominator != 0)
            {
                result = $"{numerator / denominator} {numerator % denominator}/{denominator}";
            }
            else
            {
                result = $"{numerator / denominator}";
            }
        }
        else if (denominator == numerator)
        {
            result = "1";
        }
        return result;
    }

    1 Versions
    public string Potenzieren(int a, int b, int x) // a/b ^ 2
    {
        double numerator = Math.Pow(a, x);
        double denominator = Math.Pow(b, x);

        string result = "";

        if (denominator == numerator)
        {
            result = "1";
        }
        else
        {
            result = $"{numerator}/{denominator}";
        }
        return result;
    }

    1 Versions
    public string Wurzelziehen(int a, int b) // a/b
    {
        double numerator = Math.Sqrt(a);
        double denominator = Math.Sqrt(b);

        string result = "";

        if (denominator == numerator)
        {
            result = "1";
        }
        else
        {
            result = $"{numerator}/{denominator}";
        }
        return result;
    }
}
```

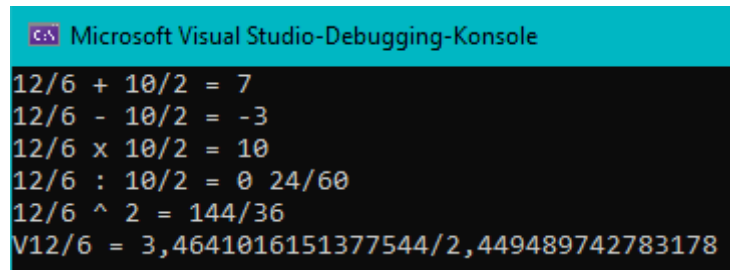
Main

```
0 Verweise
class Program
{
    0 Verweise
    static void Main(string[] args)
    {
        //test example
        Bruchrechner t = new Bruchrechner();

        int a1 = 12, a2 = 6; // 12/6
        int b1 = 10, b2 = 2; // 10/2

        Console.WriteLine("12/6 + 10/2 = " + t.Addition(a1, a2, b1, b2));
        Console.WriteLine("12/6 - 10/2 = " + t.Subtraction(a1, a2, b1, b2));
        Console.WriteLine("12/6 x 10/2 = " + t.Multiplikation(a1, a2, b1, b2));
        Console.WriteLine("12/6 : 10/2 = " + t.Division(a1, a2, b1, b2));
        Console.WriteLine("12/6 ^ 2 = " + t.Potenzieren(a1, a2, 2));
        Console.WriteLine("√12/6 = " + t.WurzelZiehen(a1, a2));
    }
}
```

Konsole



```
C:\> Microsoft Visual Studio-Debugging-Konsole
12/6 + 10/2 = 7
12/6 - 10/2 = -3
12/6 x 10/2 = 10
12/6 : 10/2 = 0 24/60
12/6 ^ 2 = 144/36
√12/6 = 3,4641016151377544/2,449489742783178
```

7) Zahlensystemumrechner

Zahlensystemumrechner Klasse

```
6 Verweise
class Zahlensystemumrechner
{
    25 Verweise
    public int Decimal_Convert<Thing>(Thing value, int num_base) //base -> decimal
    {
        return Convert.ToInt32($"{value}", num_base);
    }

    // Binaer (base 2)
    7 Verweise
    public int Bin_Convert(int value) { return Convert.ToInt32(Convert.ToString(value, 2)); } // decimal -> binary

    //Oktal (base 8)
    6 Verweise
    public string Okt_Convert(int value) { return Convert.ToString(value, 8); } // decimal -> hexadezimal

    //Hexadezimal (base 16)
    6 Verweise
    public string Hex_Convert(int value) { return Convert.ToString(value, 16); } // decimal -> hexadezimal

    1 Verweis
    public string[] Dec_to_All(int value)
    {
        int dec = value;
        int bin = Bin_Convert(value);
        string oct = Okt_Convert(value);
        string hex = Hex_Convert(value);
        return $"{dec}", $"{bin}", oct, hex;
    }
}
```

Inheritance klassen

```
2 Verweise
class Dezimal : Zahlensystemumrechner
{
    0 Verweise
    public int Addition(int x, int y) { return x + y; } // +
    0 Verweise
    public int Subtraction(int x, int y) { return x - y; } // -
    0 Verweise
    public int Multiplikation(int x, int y) { return x * y; } // *
    0 Verweise
    public int Division(int x, int y) { return x / y; } // /
}

2 Verweise
class Binaer : Zahlensystemumrechner
{
    const int num_base = 2;

    1 Verweis
    public int Addition(int x, int y) // +
    {
        int value = Decimal_Convert(x, num_base) + Decimal_Convert(y, num_base); return Bin_Convert(value);
    }

    0 Verweise
    public int Subtraction(int x, int y) // -
    {
        int value = Decimal_Convert(x, num_base) - Decimal_Convert(y, num_base); return Bin_Convert(value);
    }

    0 Verweise
    public int Multiplikation(int x, int y) // *
    {
        int value = Decimal_Convert(x, num_base) * Decimal_Convert(y, num_base); return Bin_Convert(value);
    }

    0 Verweise
    public int Division(int x, int y) // /
    {
        int value = Decimal_Convert(x, num_base) / Decimal_Convert(y, num_base); return Bin_Convert(value);
    }
}

2 Verweise
class Oktal : Zahlensystemumrechner
{
    const int num_base = 8;

    0 Verweise
    public string Addition(int x, int y) // +
    {
        int value = Decimal_Convert(x, num_base) + Decimal_Convert(y, num_base); return Okt_Convert(value);
    }

    0 Verweise
    public string Subtraction(int x, int y) // -
    {
        int value = Decimal_Convert(x, num_base) - Decimal_Convert(y, num_base); return Okt_Convert(value);
    }

    0 Verweise
    public string Multiplikation(int x, int y) // *
    {
        int value = Decimal_Convert(x, num_base) * Decimal_Convert(y, num_base); return Okt_Convert(value);
    }

    0 Verweise
    public string Division(int x, int y) // /
    {
        int value = Decimal_Convert(x, num_base) / Decimal_Convert(y, num_base); return Okt_Convert(value);
    }
}

2 Verweise
class Hexadezimal : Zahlensystemumrechner
{
    const int num_base = 16;

    0 Verweise
    public string Addition(int x, int y) // +
    {
        int value = Decimal_Convert(x, num_base) + Decimal_Convert(y, num_base); return Hex_Convert(value);
    }

    0 Verweise
    public string Subtraction(int x, int y) // -
    {
        int value = Decimal_Convert(x, num_base) - Decimal_Convert(y, num_base); return Hex_Convert(value);
    }

    0 Verweise
    public string Multiplikation(int x, int y) // *
    {
        int value = Decimal_Convert(x, num_base) * Decimal_Convert(y, num_base); return Hex_Convert(value);
    }

    0 Verweise
    public string Division(int x, int y) // /
    {
        int value = Decimal_Convert(x, num_base) / Decimal_Convert(y, num_base); return Hex_Convert(value);
    }
}
```

Main

```
0 ViewCode
static void Main(string[] args)
{

    //call out classes
    Zahlensystemumrechner conv = new Zahlensystemumrechner();

    Dezimal dec = new Dezimal();
    Binaer bin = new Binaer();
    Hexadezimal hex = new Hexadezimal();
    Oktal okt = new Oktal();

    int number = 500; //test number

    //Conversion examples
    Console.WriteLine("CONVERSION TEST");
    Console.WriteLine("\ndecimal to binary:"); //display our test number
    Console.WriteLine(number);
    Console.WriteLine(conv.Bin_Convert(number)); //convert decimal to binary

    string hex_number = conv.Hex_Convert(number);
    Console.WriteLine("\nhex to octal:");
    Console.WriteLine(hex_number);
    Console.WriteLine(conv.Okt_Convert(conv.Decimal_Convert(hex_number, 16)));

    //OR

    Console.WriteLine(); //ignore me
    foreach (string num in conv.Dec_to_All(number)) //access to all results with a loop (they are all strings)
    {
        Console.WriteLine(num);
    }

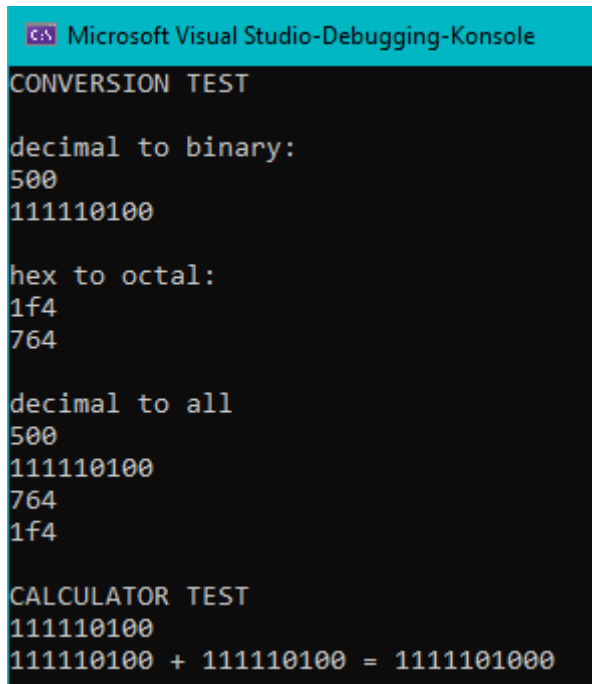
    Console.WriteLine();
    Console.WriteLine("CALCULATOR TEST");

    //Calculator examples

    int binany_number = conv.Bin_Convert(number); //we convert our decimal number to binary.
    Console.WriteLine(binany_number); //display it into the console.

    Console.WriteLine("111110100 + 111110100 = " + bin.Addition(binany_number, binany_number)); //using the Binaer class we added the first value to the other one.
}
```

Konsole



```
Microsoft Visual Studio-Debugging-Konsole

CONVERSION TEST

decimal to binary:
500
111110100

hex to octal:
1f4
764

decimal to all
500
111110100
764
1f4

CALCULATOR TEST
111110100
111110100 + 111110100 = 1111101000
```