

Angabe PLÜ

Es soll ein Teil einer Messaging-App implementiert werden. Das Programm besteht aus 4 Klassen, die zu implementieren sind. Der Aufbau der Klassen ist nachstehend im Detail beschrieben.

Ein vorgefertigtes Gerüst einer Solution steht auf MS Teams zum Download zur Verfügung. Erweitere diese Solution. Sie enthält Unit Tests mit der du deine Ergebnisse überprüfen kannst.

Klasse „SmartPhoneApp“

- Erstelle ein Default Property string *SmartPhoneId*.
- Diese Klasse soll selbst zu einer Liste werden. Leite daher von der Klasse *List* ab. **Achtung**, typisiere die Klasse *List* <>, auf den richtigen Datentyp!
- Überschreibe die *Add*-Methode. Verwende dafür in der Methodensignatur das Keyword „new“ (*public void new Add(...)*). Sie soll nur *Post* in die Liste aufnehmen können, wenn der *Post* nicht *NULL* ist. Es soll auch eine Referenz von *SmartPhoneApp* auf die Klasse *Post* gesetzt werden.
- Erstelle einen Konstruktor, der den Parameter *SmartPhoneId* entgegennimmt.
- Erstelle eine Methode *ProcessPosts*. Sie durchläuft alle *Posts* in der Liste und fügt alle HTML-Ausgaben der einzelnen *Posts* zu einem kompletten String zusammen (mittels *simplem Concatenate*) Der resultierende String soll anschließend zurückgegeben werden. (*return...*)
- Erstelle eine Methode *CalcRating*. Sie durchläuft alle *Posts* in der Liste und summiert alle *Ratings* der einzelnen *Posts* auf. Das Ergebnis soll anschließend zurückgegeben werden.
- Erstelle einen Iterator, der anhand des *Title* einen *Post* zurückliefern kann. Der Aufruf würde folgendermaßen aussehen: `posts["TextPost 6"]?.Html`. Kann der *Post* nicht in der Liste gefunden werden, soll *null* zurückgegeben werden.

Abstrakte Klasse „Post“

- Erstelle eine abstrakte Klasse „*Post*“ mit folgenden Properties:
 - *string Title* (Default Property, Read Only)
 - *DateTime Created* (Default Property, Read Only)
 - *int Rating* (Es darf nur ein Wert zwischen 1 und 5 gesetzt werden. Andernfalls soll eine *ArgumentOutOfRangeException* mit der Meldung „Range muss zwischen 1 und 5 liegen!“ geworfen werden.)
 - *abstract string Html* (Default Property, abstract, Read Only)
 - *SmartPhoneApp SmartPhone* (Default Property, Es soll eine Referenz auf die Liste beinhalten) **Achtung! Nullable-Feature:** Wähle die richtige initialisierung.
- Erstelle einen Konstruktor der die Parameter „*title*“ und „*created*“ entgegennimmt und damit die Properties befüllt. Titel darf nicht *NULL* sein. Ist das der Fall soll eine *ArgumentNullException* mit der Meldung „Titel war NULL!“ geworfen werden.
- Erstelle einen Konstruktor der den Parameter „*title*“ entgegennimmt und den vorherigen Konstruktor aufruft, um die Werte zu setzen. Für den Parameter „*created*“, verwende *DateTime.Now*.
- Überschreibe die *ToString*-Methode sodass der Inhalt des Properties *HTML* ausgegeben wird.

Klasse „TextPost“

- Sie soll von der Klasse „Post“ erben.
- Erstelle eine Klasse „TextPost“ mit folgenden Properties:
 - *string Content* (Default Property) **Achtung! Nullable-Feature:** Wähle die richtige initialisierung.
 - *int Length* (gibt die Länge des Properties Content zurück. Ist Content *NULL* soll 0 zurückgegeben werden. Verwende den null-coalesce-operator)
 - *string Html* (Überschreibt das Property der Basisklasse. Es gibt den Inhalt der Properties *Titel* und *Content* in folgender Form zurück:
„<h1>{Title}</h1><p>{Content}</p>“. Ist Content *NULL*, soll eine *ArgumentNullException* mit der Meldung "Content war NULL!" geworfen werden.)
- Die Klasse soll 2 Konstruktoren enthalten, die jeweils den entsprechenden Konstruktor der Basisklasse mit denselben Parametern aufrufen.

Klasse „ImagePost“

- Sie soll von der Klasse „Post“ erben.
- Erstelle eine Klasse „ImagePost“ mit folgenden Properties:
 - *string Url* (Default Property) **Achtung! Nullable-Feature:** Wähle die richtige initialisierung.
 - *string Html* (Überschreibt das Property der Basisklasse. Es gibt den Inhalt der Properties *Titel* und *Url* in folgender Form zurück:
„<h1>{Title}</h1>“. Ist *Url NULL*, soll eine *ArgumentNullException* mit der Meldung "Url war NULL!" geworfen werden.)
- Die Klasse soll 2 Konstruktoren enthalten, die jeweils den entsprechenden Konstruktor der Basisklasse mit denselben Parametern aufrufen.

Achte auf das *Nullable-Feature*. Ein sehr gut kann es nur ohne grüne Unterwellungen geben.

Benotung:

Pro erfolgreichem Unit-Test wird ein Punkt vergeben:

- 0 – 9 .. 5
- 10 – 11 .. 4
- 12 – 14 .. 3
- 15 – 17 .. 2
- 18 – 19 .. 1

Erwarteter Output:

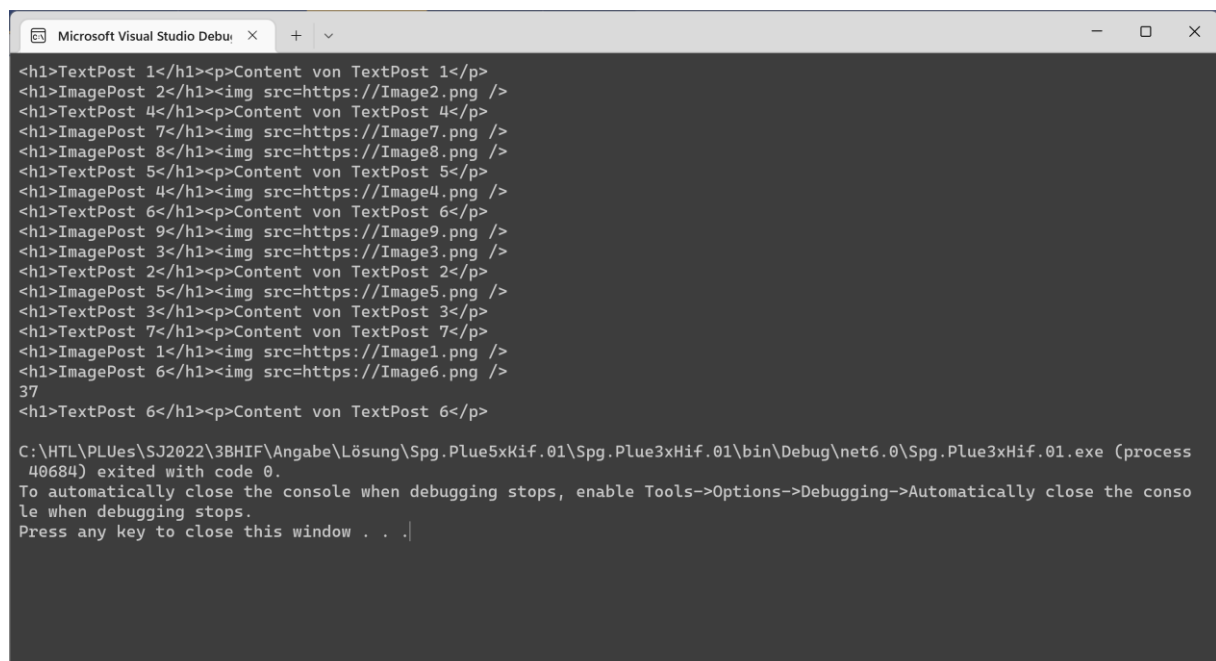
```
<h1>TextPost 1</h1><p>Content von TextPost 1</p>
<h1>ImagePost 2</h1><img src=https://Image2.png />
<h1>TextPost 4</h1><p>Content von TextPost 4</p>
<h1>ImagePost 7</h1><img src=https://Image7.png />
<h1>ImagePost 8</h1><img src=https://Image8.png />
<h1>TextPost 5</h1><p>Content von TextPost 5</p>
<h1>ImagePost 4</h1><img src=https://Image4.png />
<h1>TextPost 6</h1><p>Content von TextPost 6</p>
<h1>ImagePost 9</h1><img src=https://Image9.png />
<h1>ImagePost 3</h1><img src=https://Image3.png />
<h1>TextPost 2</h1><p>Content von TextPost 2</p>
<h1>ImagePost 5</h1><img src=https://Image5.png />
<h1>TextPost 3</h1><p>Content von TextPost 3</p>
<h1>TextPost 7</h1><p>Content von TextPost 7</p>
<h1>ImagePost 1</h1><img src=https://Image1.png />
<h1>ImagePost 6</h1><img src=https://Image6.png />
37
<h1>TextPost 6</h1><p>Content von TextPost 6</p>
```

C:\HTL\PLUes\SJ2022\3BHIF\Angabe\Lösung\Spg.Plue5xKif.01\Spg.Plue3xHif.01\bin\Debug\net6.0\Spg.Plue3xHif.01.exe (process 40684) exited with code 0.

To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.

Press any key to close this window . . .

bzw.:



```
Microsoft Visual Studio Debug Console
<h1>TextPost 1</h1><p>Content von TextPost 1</p>
<h1>ImagePost 2</h1><img src=https://Image2.png />
<h1>TextPost 4</h1><p>Content von TextPost 4</p>
<h1>ImagePost 7</h1><img src=https://Image7.png />
<h1>ImagePost 8</h1><img src=https://Image8.png />
<h1>TextPost 5</h1><p>Content von TextPost 5</p>
<h1>ImagePost 4</h1><img src=https://Image4.png />
<h1>TextPost 6</h1><p>Content von TextPost 6</p>
<h1>ImagePost 9</h1><img src=https://Image9.png />
<h1>ImagePost 3</h1><img src=https://Image3.png />
<h1>TextPost 2</h1><p>Content von TextPost 2</p>
<h1>ImagePost 5</h1><img src=https://Image5.png />
<h1>TextPost 3</h1><p>Content von TextPost 3</p>
<h1>TextPost 7</h1><p>Content von TextPost 7</p>
<h1>ImagePost 1</h1><img src=https://Image1.png />
<h1>ImagePost 6</h1><img src=https://Image6.png />
37
<h1>TextPost 6</h1><p>Content von TextPost 6</p>

C:\HTL\PLUes\SJ2022\3BHIF\Angabe\Lösung\Spg.Plue5xKif.01\Spg.Plue3xHif.01\bin\Debug\net6.0\Spg.Plue3xHif.01.exe (process 40684) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Viel Glück und gutes Gelingen!!