

Name: _____

Matrikelnummer: _____ Studienkennzahl: _____

1. Die .NET-Architektur (6 + 4 = 10 Punkte)

- a. Microsoft setzt mit der Einführung der CLR (Common Language Runtime) genau wie Sun mit der JVM (Java Virtual Machine) auf eine zusätzliche Ebene zwischen Hochsprachen- und Maschinenprogramm. Welche Vorteile bringt diese Aufspaltung der Übersetzung in zwei Phasen (d.h. Übersetzung der Quellsprache in Zwischencode und dann Übersetzung des Zwischencodes in ein Maschinenprogramm)? Begründen Sie Ihre Antworten kurz.

__ / 6

- b. Welche Informationen stecken in der so genannten "Metadata"? Wo befinden sie sich? Wann und wie gelangen sie dorthin? Welche Vorteile hat dieser Ansatz gegenüber älteren Technologien, wie IDL?

__ / 4

2. Klassenbibliothek: XML (4 + 2 + 4 = 10 Punkte)

- a. *Collections*: Geben Sie wichtige Interfaces und Klassen der Collection-Frameworks von .NET an und erläutern Sie ihre Bedeutung.

__ / 4

- b. *Reflection*: Wozu dienen die Typen im Namensraum System.Reflection.Emit?

__ / 2

- c. *XML*: Erläutern Sie einige XML-basierte Standards, welche im .NET-Framework implementiert sind. Welche Aufgaben werden durch sie gelöst?

__ / 4

3. ASP.NET (20 Punkte)

Jemand hat folgende Webseite als Anmeldeformular für ein Seminar entworfen:

Name	Schmidt
Seminar (300 Euro)	<input checked="" type="checkbox"/>
Workshop (100 Euro)	<input type="checkbox"/>
Gesamtpreis	300 Euro

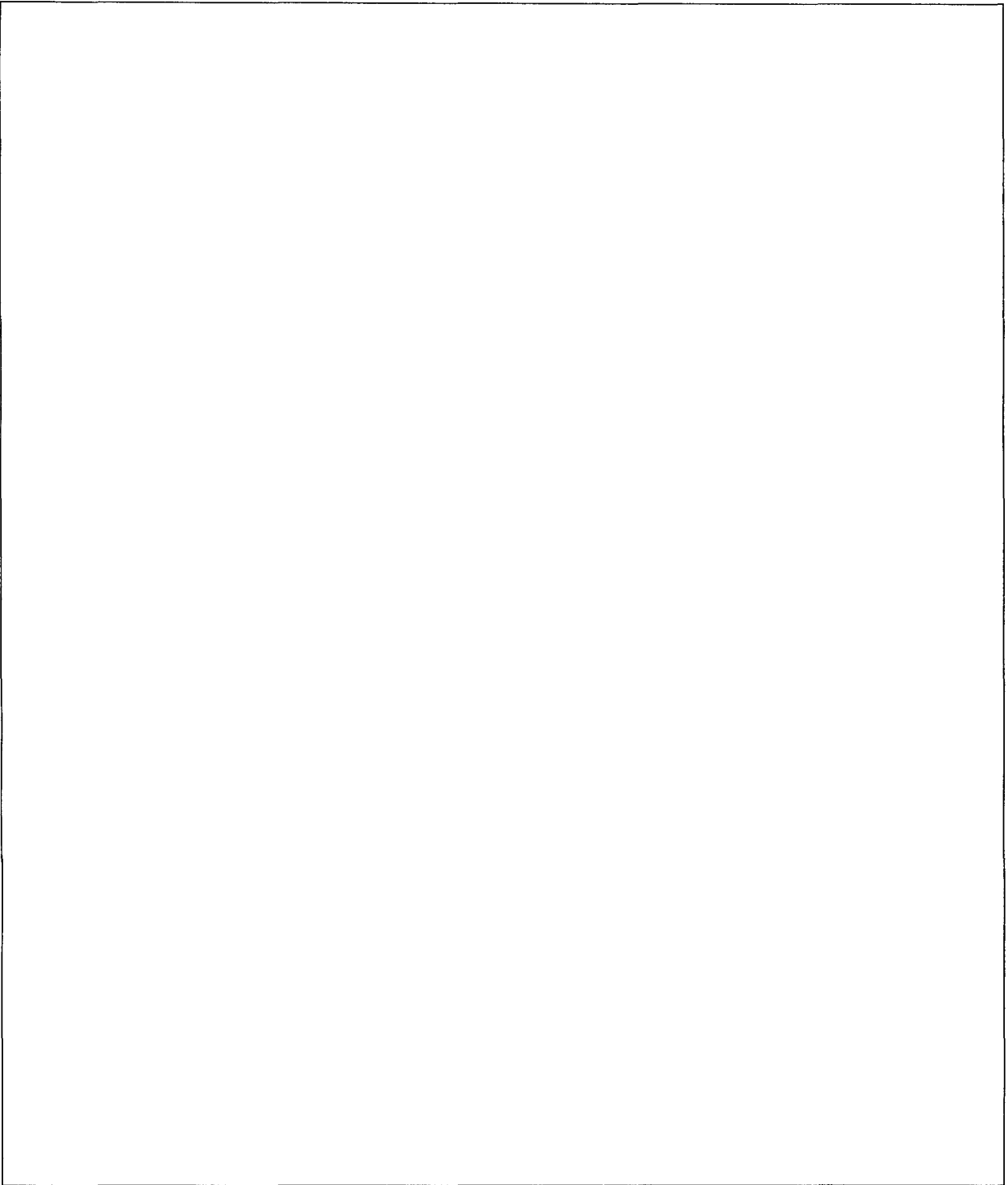
Anmelden

Die.aspx-Seite dazu sieht folgendermaßen aus:

```
<%@ Page Language="C#" Inherits="MyPage" Src="Klausur02a.aspx.cs"%>
<html>
  <body>
    <h2>Anmeldeformular</h2>
    <form Runat="server">
      <table border="1">
        <tr>
          <td>Name</td>
          <td><asp:TextBox ID="name" Runat="server"/>
        </tr>
        <tr>
          <td>Seminar (300 Euro)</td>
          <td><asp:CheckBox ID="seminar" Runat="server"/>
        </tr>
        <tr>
          <td>Workshop (100 Euro)</td>
          <td><asp:CheckBox ID="workshop" Runat="server"/>
        </tr>
        <tr>
          <td>Gesamtpreis</td>
          <td><asp:Label ID="price" Runat="server"/> Euro
        </tr>
      </table>
      <br>
      <asp:Button ID="register" Text="Anmelden"
        OnClick="Register" Runat="server"/>
    </form>
  </body>
</html>
```

Schreiben Sie den nötigen Hintergrundcode, um den Gesamtpreis aus den Checkboxes für "Seminar" und "Workshop" zu berechnen. Die Anmeldung muß nicht abgespeichert werden, sondern es soll nur der Gesamtpreis berechnet und im Label *price* angezeigt werden.

Antwort 3.



 / 20

4. ADO.NET (10 + 10 = 20 Punkte)

Im Folgenden sind zwei unvollständige ADO.NET-Programme gegeben. Vervollständigen Sie diese Programme.

a. Verbindungsorientierter Zugriff mit DataReader: Folgende Programmschritte sind zu ergänzen:

- Kommandoobjekt erzeugen und ausführen
- über alle Datenzeilen iterieren
- Werte der Spalten CompanyName und ContactName auf der Konsole ausgeben.

```
using System;
using System.Data;
using System.Data.OleDb;

public class ADOTestExample {

    public static void Main(string[] args) {

        string connString = "Provider=Microsoft.Jet.OLEDB.4.0;" +
            "Data Source=Customers.mdb");
        string commandString = "SELECT CompanyName, ContactName FROM Customers";
        IDbCommand cmd;
        IDataReader reader;
        OleDbConnection con;
        try {
            con = new OleDbConnection(connString);
            con.Open();

            // Antwort (a) →
```

b. Verbindungsloser Zugriff mit DataAdapter und DataSet: Folgende Programmschritte sind zu ergänzen:

- DataSet füllen
- über alle Zeilen der Customer-Tabelle iterieren
- Werte der Spalten CompanyName und ContactName auf der Konsole ausgeben.

```
using System;
using System.Data;
using System.Data.OleDb;

public class ADOTestExample {

    public static void Main(string[] args) {

        string connString = "Provider=Microsoft.Jet.OLEDB.4.0;" +
            "Data Source=Customers.mdb");
        string commandString = "SELECT CompanyName, ContactName FROM Customers";
        OleDbDataAdapter adapter;
        DataSet ds;

        adapter = new OleDbDataAdapter(commandString, connectionString);

        // Antwort (b) →
```

Antwort 4 a.

___ / 10

```
    } catch (Exception e) {  
        Console.WriteLine(e.Message);  
    } finally {  
        try {  
            if (con != null)  
                // Verbindung schließen  
                con.Close();  
        } catch (Exception ex) { Console.WriteLine(ex.Message); }  
    }  
}
```

Antwort 4 b.

___ / 10