Кировское областное государственное образовательное

бюджетное учреждение среднего профессионального образования

"Кировский авиационный техникум"

Отчет по производственной практике

09.02.07.ПП02ИР31.021Д1

Студентка группы ИР-31

Кокорина В. В.

Руководитель практики от техникума

Ардашева Ю. М.

1. Инспекция программного кода на предмет соответствия стандартам кодирования
   1. Исходный код

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{ Tperson resp;

class Tperson

{

public string fname;

public string fsome;

public int salary;

public Tperson ( String fam, string name, int okl)

{

fsome = fam;

fname = name;

salary = okl;

}

public virtual string info()

{

return fsome + ' ' + fname;

}

public virtual double GetSum()

{

return salary;

}

}

class TStud : Tperson

{

public double fGr;

public DateTime fGod;

public TStud(String fam, string name, int okl, DateTime god, double gr)

:base(fam,name,okl)

{

fGr = gr;

fGod = god;

}

public override string info()

{

return (string.Format("Студент {0}{1}{2} , Дата рождения {3}, средний бал {5}, стипендия {4}",

fsome, ' ', fname, fGod.ToString("D"), Convert.ToDouble(GetSum()),

Convert.ToDouble(fGr)));

}

public override double GetSum()

{

if (fGr >= 4.5) return salary \* 2;

if ((fGr >= 3.5) && (fGr<4.5)) return salary + (salary / 2);

if (fGr < 3) return salary = 0;

else return salary;

}

}

class TProf : Tperson

{

public string fkat;

public string fdep;

public TProf(String fam, string name, int okl, string kat, string dep)

: base(fam, name, okl)

{

fkat = kat;

fdep = dep;

}

public override string info()

{

return (string.Format("Преподаватель {0}{1}{2} , Кафедра {3}, категория {5}, оклад {4}",

fsome, ' ', fname, fdep , Convert.ToDouble(GetSum()), fkat));

}

public override double GetSum()

{

if (fkat == "высшая") return salary + (salary \* 0.7);

if (fkat == "первая") return salary + (salary \* 0.4);

if (fkat == "без категории") return salary;

else return salary;

}

}

public Form1()

{

InitializeComponent();

}

private void label5\_Click(object sender, EventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

Close();

}

private void button1\_Click(object sender, EventArgs e)

{

if (radioButton1.Checked)

{

resp = new TStud ("Петров", "Петр", 1000, Convert.ToDateTime("12.01.2000"),3.6);

textBox5.AppendText(resp.info() + "\n");

}

if (radioButton2.Checked)

{

resp = new TProf("Иванов", "Иван", 5000, "высшая", "математика");

textBox5.AppendText(resp.info() + "\n");

}

}

}

}

* 1. Стили наименования

Основным стилем в C# является стиль Camel case.

Camel case – стиль написания составных слов, при котором несколько слов пишутся слитно без пробелов, при этом каждое слово внутри фразы пишется с прописной буквы. Первая буква первого слова в идентификаторе в нижнем регистре, все первые буквы последующих слов – в верхнем.

* 1. Использование верхнего и нижнего регистра в именах

Запрещается создавать два различных namespace’а, функции, типа или свойства с одинаковыми именами, отличающиеся только регистром. Запрещается создавать функции с именами параметров, отличающимися только регистром.

* 1. Общие правила именования идентификаторов

При именовании идентификаторов не используются аббревиатуры или сокращения, если только они не являются общепринятыми.

Если имя идентификатора включает в себя сокращение – сокращение пишется в upper case. Исключение - когда имя идентификатора должно быть указано в camel case и сокращение стоит в начале имени идентификатора. В этом случае сокращение пишется в нижнем регистре.

* 1. Правила именования классов

Следует избегать имен классов, совпадающих с именами классов .NET Framework;

Для классов используется стиль именования pascal case;

Для классов, унаследованных от CollectionBase используется суффикс Collection, перед которым указывается тип объектов, для которых используется коллекция.

В качестве имен классов используются имена существительные;

Имя класса не должно совпадать с именем namespace’а. Изм. Лист № докум. Подпись Дата Лист 6 09.02.07.ПП02ИР31.021Д1

Если класс представляет собой сущность, хранимую в базе данных – имя класса соответствует имени таблицы. В этом случае имя класса – это название сущности в единственном числе, имя таблицы – во множественном числе.

При создании классов потомков их имена состоят из имени базового класса и суффикса класса потомка, если суффиксов несколько – они разделяются символом подчеркивания.

Имена файлов, в которых находятся классы, совпадают с именами классов. Для именования файлов используется стиль pascal case.

Исправления в коде:

TStud - TpersonStud

TProf – TpersonProf

* 1. Правила именования интерфейсов

Имена интерфейсов начинаются с буквы I, после которой следует название интерфейса в pascal case.

* 1. Правила именования generic’ов

Generic’и обозначаются буквой T, если generic’ов несколько их имена начинаются с буквы T.

* 1. Правила именования функций

Имена функций должны давать четкое представление о том, какое действие эта функция выполняет. Имя функции начинается с глагола, указывающего на то, какое действие она выполняет;

Большие функции, не умещающиеся на одном экране, делятся на несколько private функций меньшего размера, имена таких вспомогательных функций состоят из имени основной (большой) функции и существительного, глагола или фразы, которые уточняют действие вспомогательной функций, разделенные подчеркиванием. Основная и вспомогательная функции объединяются в регионы. Вспомогательные функции вызываются только из основной функции.

* 1. Правила именования параметров функций

Для именования параметров используется стиль camel case;

Имена параметров должны давать четкое представление о том, для чего используется параметр, и какое значение следует передать при вызове функции.

* 1. Правила именования переменных

Для именования переменных используется стиль camel case;

Переменные объявляются согласно следующему шаблону:

<Тип> <Название поля>;

Вывод: не правильно даны названия классам TStud и TProf.