

EPAM Systems, RD Dep.
Homework Task

06 – Basics of Object-Oriented Programming

REVISION HISTORY					
Ver.	Description of Change	Author	Date	Approved	
				Name	Effective Date
1.0	Initial version	Alexander_Kuznetsov1	20.01.2012		
2.0		Nikolay Piskarev	29.05.2017		

Наследование

Задание 1

На основе класса User (см. задание 3 из предыдущей темы), создать класс Employee, описывающий сотрудника фирмы. В дополнение к полям пользователя добавить поля «стаж работы» и «должность». Обеспечить нахождение класса в заведомо корректном состоянии.

Задание 2

Создать класс Ring (кольцо), описываемое координатами центра, внешним и внутренним радиусами, а также свойствами, позволяющими узнать площадь кольца и суммарную длину внешней и внутренней границ кольца. Обеспечить нахождение класса в заведомо корректном состоянии.

Полиморфизм

Задание 3

Напишите заготовку для векторного графического редактора. Полная версия редактора должна позволять создавать и выводить на экран такие фигуры как: Линия, Окружность, Прямоугольник, Круг, Кольцо. Заготовка, для упрощения, должна представлять собой консольное приложение с функционалом:

1. Создать фигуру выбранного типа по произвольным координатам.
2. Вывести фигуры на экран (для каждой фигуры вывести на консоль её тип и значения параметров).

Задание 4

Создайте иерархию классов и пропишите ключевые методы для компьютерной игры (без реализации функционала). Суть игры:

1. Игрок может передвигаться по прямоугольному полю размером Width на Height.
2. На поле располагаются бонусы (яблоко, вишня и т.д.), которые игрок может подобрать для поднятия каких-либо характеристик.

3. За игроком охотятся монстры (волки, медведи и т.д.), которые могут передвигаться по карте по какому-либо алгоритму.

4. На поле располагаются препятствия разных типов (камни, деревья и т.д.), которые игрок и монстры должны обходить.

Цель игры – собрать все бонусы и не быть «съеденным» монстрами.

Требования к оформлению

1. Для каждой домашней работы необходимо создать собственную папку в корне репозитория.
2. Каждое задание должно представлять собой отдельный проект (project) в рамках общего решения (solution).
3. Правила именования:

Все названия и имена должны быть переведены на английский язык. Именованье транслитерацией не допускается.

 - a. Маска именования репозитория: EPAM_Ext_Lab_<Квартал старта группы>_<Год>_<Фамилия_Имя>.
Например, EPAM_Ext_Lab_Q2_2017_Ivanov_Ivan
 - b. Маска именования решения: HMT_<Порядковый номер>.
Например, HMT_01, HWT_11.
 - c. Маска именования проектов: <Задача><Номер задачи>.
Например, Task01, Task12.
 - d. Каждое задание должно быть оформлено следующим образом:
 1. Формулировка задания в виде комментария (просто копировать-вставить из этого файла).
 2. Код, решающий задачу.
4. Написать письмо преподавателю, информирующее о том, что домашняя работа выполнена. Тема письма должна выглядеть так: [STYYYY_N]Фамилия_Имя_TaskNN. Здесь ST – сокращённое наименование программы (Students Training), YYYY – номер года, N – номер группы, NN – порядковый номер задания в виде двух цифр, напр. 01, 05, 12 и т.д.
Например, ST2017_1 Иванов Иван Task01.

Рекомендации к оформлению кода

1. При добавлении нового файла желательно вверху файла добавить блок с комментарием, для чего предназначен этот файл.

2. При добавлении метода в блоке `<summary></summary>` описать предназначение метода, его входных/выходных параметров и возвращаемых значений.

3. Использовать утилиту StyleCop для проверки правил оформления кода. Перед отправкой задания на проверку запускать StyleCop и исправлять все предупреждения, возникающие в секции Warning Visual Studio. Предупреждения об отсутствии документации можно игнорировать (либо настроить StyleCop таким образом, чтобы он не проверял документирование кода).