Задачи с созданием классов(MathUtils, Counter, StringUtils, Circle)

**Вопрос 1: Почему программа не компилируется?**

Ошибка компиляции возникает из-за уровня доступа. Класс Person имеет модификатор internal, а класс Employee — public. В C# нельзя наследовать internal-класс от public-класса, так как это нарушает принцип доступности: производный класс не может быть доступнее базового.

**Вопрос 2: Какие конструкторы вызываются?**

Сначала вызывается конструктор базового класса Person(

Далее вызывается:

public Person(string name, int age)

Этот инициализирует поля name и age.

После этого возвращаемся в конструктор Employee, который инициализирует company.

Итого порядок вызова:

Employee(string, string)

Person(string)

Person(string, int)

**Вопрос 3: Как запретить наследование от класса?**

Использовать модификатор sealed

**Вопрос 4: Что выведет программа?**

Ошибка компиляции: базовый класс Auto не имеет конструктора по умолчанию, а в Truck конструкторе нет вызова base(seats).

**Вопрос 5: Что выведет программа?**

При создании new Truck вызываются:

Сначала конструктор базового класса Auto():

Затем конструктор Truck(decimal):

Вывод:

Auto has been created

Truck has been created

Truck with capacity 1.1

**Вопрос 6: Что выведет программа?**

Порядок событий:

В конструкторе Person(string) присваивается Name = "Tim", независимо от параметра.

Name = "Sam" в инициализаторе перезаписывает значение.

Все промежуточные значения (Ben, Tim, Bob, Tom) не видны, потому что:

"Ben" — начальное значение, перезаписывается в конструкторе.

"Tim" — задаётся конструктором Person.

Name = "Sam" задаётся после выполнения конструктора.

Вывод:

Sam