

2.2. Интерфейс PriceFormatter

В данном задании рассмотрена реализация интерфейса. Интерфейс – конструкция, которая позволяет определить необходимую для определённой цели часть структуры класса. Реализация интерфейса заключается в создании класса с использованием этого интерфейса. В данном случае реализуется интерфейс PriceFormatter, который возвращает цену товара в нужном формате. Код для тестирования приведён на листинге 4.

Листинг 4 – Код для тестирования интерфейса PriceFormatter

```
class ExampleUnitTest
{
    @Test
    fun example()
    {
        val iphoneCase = Product(price = 123.5, discountPercent = 30)
        val priceFormatter: PriceFormatter = // название вашего класса
        val discountIphoneCasePrice = iphoneCase.calcDiscountPrice()
        println(priceFormatter.format (discountIphoneCasePrice))
    }
}

class Product(
    private val price: Double,
    private val discountPercent: Int = 0
)
{
    fun calcDiscountPrice(): Double = price * (1 - discountPercent / 100.0)
}

interface PriceFormatter
{
    fun print(price: Double): String
}
```

К примеру, для форматирования числа можно использовать класс, приведённый на листинге 5.

Листинг 5 – Пример класса форматирования цены

```
class MyPriceFormatter : PriceFormatter
{
    override fun format(price: Double): String
    {
        if (price % 1 == 0.0)
        {
            return price.toInt().toString()
        } else
        {
            return String.format("%.2f", price)
        }
    }
}
```

При использовании этого класса число будет отображено как целое, если его дробная часть равна нулю, в противном случае оно будет округлено до двух знаков после запятой.