

- Если вам не удалось выполнить задание, сравните получившийся у вас код класса MoneyCalculator с эталонным кодом:

```
public class MoneyCalculator {
    private static final int BANKNOTES_PER_PACK = 100;
    private static final int PACKS_IN_CASSETTE = 20;

    public static int calculateCassetteCount(
        int banknoteCount
    ) {
        int packsCount = banknoteCount /
            BANKNOTES_PER_PACK +
            (banknoteCount %
            BANKNOTES_PER_PACK == 0 ? 0 : 1);
        int cassetteCount = packsCount /
            PACKS_IN_CASSETTE +
            (packsCount %
            PACKS_IN_CASSETTE == 0 ? 0 : 1);
        return cassetteCount;
    }
}
```

- Внесите в свой код исправления таким образом, чтобы он точно соответствовал эталонному коду выше.
- Пример кода для проверки корректности работы метода calculateCassetteCount:

```
System.out.println(
    MoneyCalculator.calculateCassetteCount(1999)
);

System.out.println(
    MoneyCalculator.calculateCassetteCount(2000)
);

System.out.println(
    MoneyCalculator.calculateCassetteCount(2001)
);
```

- Запустите свой код и убедитесь, что он работает. Для примера выше в консоль должен выводиться следующий результат:

```
1
1
2
```

- Теперь вы знаете, как работает и применяется на практике оператор получения остатка от деления (%).