

Mãos na massa: Wrappers

1) Dentro do pacote `br.com.bytebank.test.util`, crie uma nova classe `TesteWrapperInteger` com o método `main`. Nela, teste o *autoboxing/unboxing* usando o *wrapper* `java.lang.Integer`:

```
//package e imports omitidos

public class TesteWrapperInteger {

    public static void main(String[] args) {

        Integer idadeRef = 29; //autoboxing, é criado um objeto do tipo Integer
        int primitivo = new Integer(21); //que locura!! unboxing

        List<Integer> lista = new ArrayList<>();
        lista.add(idadeRef); //ok
        lista.add(primitivo); //autoboxing

        int i1 = lista.get(0); //unboxing
        Integer i2 = lista.get(1); //ok

        System.out.println(i1); //29
        System.out.println(i2); //21

    }
}
```

2) Agora vamos praticar alguns métodos da classe `Integer`. Adicione no final do método `main`:

```
Integer valorRef = Integer.valueOf(33); // delegando a criação para método valueOf
int valorPri = valorRef.intValue(); //desembrulhando, pegando o valor primitivo do objeto wrapper

System.out.println(valorPri); //33
```

3) Agora vamos testar o *parsing*, que é a conversão do `String` para um tipo específico. No fim do método `main`, adicione:

```
Integer iParseado1 = Integer.valueOf("42"); //parseando e devolvendo referencia
int iParseado2 = Integer.parseInt("44"); //parseando e devolvendo primitivo

System.out.println(iParseado1); //42
System.out.println(iParseado2); //42
```

4) Verifique as constantes da classe `Integer`. Para isso, adicione no `main`:

```
System.out.println(Integer.MAX_VALUE); // 2^16
System.out.println(Integer.MIN_VALUE); //-2^16

System.out.println(Integer.SIZE); // 4 bits
System.out.println(Integer.BYTES); //4 Bytes
```

Rode o código para ver a saída. Tudo deve estar funcionando.