

**Bài 1:** Nếu một phương thức có giá trị trả về khác nhau có phải là overloading không?

➔ Có

Overloading: Hai hay nhiều phương thức có cùng tên trong một class.

Để xét độ chính xác của đề bài, ta thêm một số phương thức từ ví dụ của slide, ta thêm thêm `String setDate` và `int setDate`, thì kết quả chương trình vẫn chạy bình thường

Kết quả chương trình

```
void setDate(int month, int day, int year)
void setDate(String month, int day, int year)
void setDate(int year)
String setDate(int month, int year)
int setDate(String month, int year) 20
Press any key to continue . . . ■
```

Code

```
public class App {
    public void setDate(int month, int day, int year) {
        System.out.println("void setDate(int month, int day, int year)");
    }
    public void setDate(String month, int day, int year) {
        System.out.println("void setDate(String month, int day, int year)");
    }
    public void setDate(int year) {
        System.out.println("void setDate(int year)");
    }
    public String setDate(int month, int year) {
        return "String setDate(int month, int year)";
    }
    public int setDate(String month, int year) {
        return 20;
    }
    public static void main(String[] args) {
        App myObj = new App();
        myObj.setDate(10, 20, 2021);
        myObj.setDate("Oct", 10, 2021);
        myObj.setDate(2000);
        System.out.println(myObj.setDate(10, 2021));
        System.out.println("int setDate(String month, int year) " + myObj.setDate("Oct", 2021));
    }
}
```

(code tô vàng là code mới thêm để kiểm tra tính overloading)