```java
public void myMethod(int schoolClassNumber) {

    int noOfStudents = getStudentCapacityofClass(schoolClassNumber);
    String[] names = new String[noOfStudents];
    names[0] = "new value";
}
```

Without Exception handling, we have to do this:
notice 2 things : readability and error code need to be returned

```java
public  int myMethod(int schoolClassNumber) {

    int errorCode = 0; // 0 means no success

    if (schoolClassNumber > 0 && schoolClassNumber <= 12) {
        int noOfStudents = getStudentCapacityofClass(schoolClassNumber);
        if (noOfStudents != 0) {
            String[] names = new String[noOfStudents];
            if (names != null && names.length > 0) {
                names[0] = "new value";
            } else {
                return -3;
            }
        } else {
            return -2;
        }
    } else {
        errorCode = -1;
    }
    return errorCode;
}
```

```java
public void myMethod(int schoolClassNumber) {

    try {
        int noOfStudents = getStudentCapacityofClass(schoolClassNumber);
        String[] names = new String[noOfStudents];
        names[0] = "new value";
    }
    catch (IndexOutOfBoundsException expObj){
        //do something
    }
    catch (Exception expObj){
        //do something
    }
}
```

```java
public int myMethod(int a, int b) {
    int val;
    try{
        val = a/b;
    } catch (ArithmeticException exp){
        val = -1;
    }
    return val;
}

public int myMethod2(int a, int b) {

    if(b == 0){
        return -1;
    }

    int val = a/b;
    return val;
}
```

Try to avoid using exception handling, if you can. Like in this example, its not required